ENHANCEMENT OF AEROELASTIC ROTOR AIRLOAD PREDICTION METHODS

A Dissertation
Presented to
The Academic Faculty

By

Jennifer N. Abras

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy in Aerospace Engineering

Georgia Institute of Technology
May 2009

ENHANCEMENT OF AEROELASTIC ROTOR AIRLOAD PREDICTION METHODS

Approved by:

Dr. Marilyn Smith, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Olivier Bauchau
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Mark Costello
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Stephen Ruffin
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Marvin Moulton
Aeromechanics Division
US Army Aviation Engineering Directorate, AMRDEC

Date Approved: March 31, 2009

ACKNOWLEDGEMENTS

The main focus of this research effort is to improve existing computational prediction methods. Thus many of the components of this research are well developed codes which have taken years of independent research to develop. I would like to thank all those who contributed their computational codes and resources as well as those who contributed their time and advice to this research effort. First, I would like to thank Prof. Olivier Bauchau for allowing the use of his comprehensive code DYMORE and for his advice in this area. I would also like to thank his students, Haying Liu and Afifa Zaki, for their contributions to improving the capabilities of DYMORE by implementing the external coupling and aerodynamic computations. I would like to thank the FUN3D development team for allowing the use of their CFD code, especially Bob Biedron, Beth Lee-Rausch, Mike Park, and Karen Bibb for their specific contributions to elastic motion and grid adaptation. I would like to thank Prof. Gordon Leishman for contributing the free wake code MFW, and Dr. Alan Egolf for sharing the benefit of his experience with the MFW code. I would also like to thank my fellow graduate research assistants Eric Lynch, Nicholas Burgess, and Chirag Talati for their contributions in providing OVERFLOW data for comparison with the new methods, and grid generation. I would like to acknowledge the valuable vgrid advice provided by Ryan Czerwiec of NAVAIR. I would like to acknowledge the computer resources used during this project, which were provided by the DoD High Performance Computing Modernization Program at the Naval Oceanographic Office (NAVO), US Army Engineer Research and Development Center (ERDC), Army Research Laboratory (ARL), and Maui High Performance Computing Center (MHPCC). I would especially like to thank my S/AAA Roger Strawn for his invaluable support in managing the accounts for the HPC resources. I would also like to thank the committee members Prof.Olivier Bauchau, Prof.

Mark Costello, Prof. Stephen Ruffin, Dr. Marvin Moulton, and Dr. David O'Brien Jr. for contributing their advice and their time during this process. Last but not least, I would like to thank my advisor Prof. Marilyn Smith, without whom none of this would have been possible.

This research was partially funded by the Government under Agreement No. W911W6-05-02-0003. The Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation thereon. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Aviation and Missile Research, Development and Engineering Center or the Government. This research was also partially funded under VLRCOE contract W911W6-06-2-00004 and RCOE grant NGT 2-52274. Contract monitor Dr. Mike Rotkowski of the AFDD Ames Research Center.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

A	Cross-sectional area, m^2
a	Speed of sound, m/s
a_0	Lift curve slope, $1/deg$
b	Semi-chord, m
c_d	Measured non-dimensional static airfoil drag
c_{d0}	Zero lift airfoil drag coefficient
c_l	Measured static airfoil lift coefficient
c_m	Measured static airfoil pitching moment coefficient
c_{m0}	Zero lift airfoil pitching moment coefficient
D	Total airfoil drag, N
$E[\ ]$	Inflow acceleration operator
J	Jacobian matrix for trim
L	Total airfoil lift, N
$L[\ ]$	Quasi-steady inflow operator
$M_{c/4}$	Total airfoil positive nose up pitching moment, Nm
M	Mach number
n	Number of rotor blades
r	Position of a wake marker and radial location, m
R	Rotor radius, m
s	Curvilinear radial distance, m
t	Time, s
T	Time constant for auto-pilot trim, s
$\hat{u}$	Output parameter
(u, v, w)	Local velocity components, m/s
V	Local total velocity, m/s

$\bar{w}$	Normal induced velocity, positive down, non-dimensionalized by ΩR
x, z	Cartesian coordinate system in the stream and normal directions
x_{ac}	Non-dimensional location of the aerodynamic center of an airfoil section
y	Input parameter
α	Angle of attack
α_{eff}	Effective angle of attack
α_f	Effective angle of attack which gives the equivalent leading edge unsteady pressure as the quasi-steady case
β	Prandtl-Glauert compressibility correction
Γ	Blade circulation
$\Delta(.)$	Change in a variable
ϵ	Strain
η	Viscosity recovery factor ≈ 0.95
λ	Tip loss correction factor, non-dimensionalized by ΩR
ν	Kinematic viscosity, m^2/s
ρ	Density, kg/m^3
Φ	Pressure function, non-dimensionalized by $\rho\Omega^2 R^2$
ψ	Azimuth angle, rad
ω	Vorticity, m^2/s
Ω	Rotor rotational speed, rad/s
ξ	Wake or vortex age, rad
$(.)^*$	Derivative with respect to non-dimensional time
$\vec{(.)}$	Vector quantity
$\bar{(.)}$	Non-dimensional quantity
$(.)_b$	Bound
$(.)_f$	Final time step
$(.)_i$	Initial time step

$(.)_j$	Loop parameter
$(.)_m$	Current time step
$(.)_{nw}$	Near wake
$(.)_o$	Target
$(.)_t$	Blade tip
$(.)_\infty$	Freestream
6DoF	Six Degree of Freedom
CFD	Computational Fluid Dynamics
CSD	Computational Structural Dynamics
DiRTLib	Donor interpolation Receptor Transaction Library
DLR	Deutsches Zentrum für Luft- und Raumfahrt
FSI	Fluid Structure Interation
FW	Free Wake
FUN3D	Fully Unstructured Navier-Stokes 3D
ICM	Influence Coefficient Matrix
LMP	Low Mach Number Preconditioning
MFW	Maryland Free Wake
MPI	Message Passing Interface
ONERA	Office National d'Etudes et de Recherches Aérospatiales.
RANS	Reynolds Averaged Navier-Stokes
RBM	Rigid Body Motion
SUGGAR	Structured, Unstructured, and Generalized overset Grid AssembleR
URANS	Unsteady Reynolds Averaged Navier-Stokes

SUMMARY

The accurate prediction of rotor airloads is a current topic of interest in the rotorcraft community. The complex nature of this loading makes this problem especially difficult. Some of the issues that must be considered include transonic effects on the advancing blade, dynamic stall effects on the retreating blade, and wake vortex interactions with the blades, fuselage, and other components. There are numerous codes to perform these predictions, both aerodynamic and structural, but until recently each code has refined either the structural or aerodynamic aspect of the analysis without serious consideration to the other, using only simplified modules to represent the physics. More recent work has concentrated on coupling CFD and CSD computations to be able to use the most accurate codes available to combine the best of the structural and the aerodynamic codes. However, CFD codes are the most computationally expensive codes available, and although combined CFD and CSD methods are shown to give the most accurate predictions available today, the additional accuracy must be deemed worth the time required to perform the computations.

The objective of the research is to both evaluate and extend a range of prediction methods comparing accuracy and computational expense. This range covers many methods where the highest accuracy method shown is a delta loads coupling between an unstructured CFD code and a comprehensive code, and the lowest accuracy is found through a free wake and comprehensive code coupling using simplified 2D aerodynamics. From here, methods to improve the efficiency and accuracy of the CFD code are considered through implementation of grid adaptation and low Mach number preconditioning methods. Applying grid adaptation allow coarser grids to be used where high gradients in the physics are not present, reserving the denser areas for more interesting regions. For steady-state problems, clustering of the grid provides better wake resolution behind the actuator disk.

This method is proven to work for the steady-state equations, but its application to rotor flows using the time-accurate equations still needs to be tested. Low Mach number preconditioning is also an efficiency and an accuracy improvement which allows the CFD code to work for a wider range of Mach numbers within a single simulation. There are many cases, especially for rotor flows, where the range of Mach numbers contained in the flow field cover both the incompressible and compressible regimes. Thus, applying the compressible equations to the entire flow field results in governing equations with high stiffness matrices. The preconditioning reduces the numerical stiffness and thus improves the quality of the results. This improved quality is demonstrated through low speed rotor-fuselage simulations.

Further efficiency improvements are obtained by modifying the codes used in the analysis to include more simplified methods. On the aerodynamic side, a coupling between a CFD code and a prescribed rigid motion module has been completed, and on the structural side a coupling between a CSD code and a combination of a 2D airfoil theory and a free wake code is shown. It is found that the rigid motion method is more appropriately applied where blade elasticity is not significant, and the CSD method is far more efficient than CFD methods, but with a penalty in accuracy. The exact formulation of the 2D aerodynamic model used in the CSD code is discussed, as are efficiency improvements to improve the speed of the free wake code. The advantages of the computationally expensive free wake code are tested against a faster dynamic inflow model, and show that there are improvements when using the more accurate wake formulation. A comparison of these methods evaluates the advantages and consequences of each combination, including the types of physics that each method is able to, or not able to, capture through examination of how closely each method matches flight test data.

CHAPTER 1

INTRODUCTION

1.1 Motivation

Predicting rotor airloads accurately has been a task of interest for the past 20-30 years. The inherent complexity of the rotor aerodynamics and dynamics make this problem especially difficult, as illustrated in Figure 1. As each blade rotates, it trails a wake behind it. This wake takes the form of a sheet trailing behind each blade, but most of the vorticity is concentrated at the tip of the blade, where the higher pressures on the bottom of the blade and the lower pressures on the top of the blade cause the flow to roll over the blade tip. The resulting structure of this wake as it propagates further from the origin becomes approximately helical because of blade rotation. This wake interacts with the fuselage of the helicopter as well as the tail and tail rotor, and can also interact with the main rotor blades under certain flight conditions. In addition to this complex wake, the local flow velocity changes with azimuth location. These changes in velocity are greatest on the advancing blade, which can result in transonic flow and shock formation. The retreating blade sees the lowest velocities possibly resulting in blade stall and flow reversal. These phenomena can create a condition of dynamic stall on the rotor blades whose behavior is more complex than that of static stall [4].

While the rotor aerodynamics contain a wealth of complex phenomena, the dynamics of the blades also add additional complexity. As the blades rotate, there are coupled flapping, pitching, and lead-lag motions controlling the rigid blade motions. In addition to these rigid motions there are also elastic blade deformations. All of these motions combined change the position of the blade and thus the flow seen by the blade. Therefore, an accurate prediction of the rotor airloads relies on the accurate prediction of all three components,

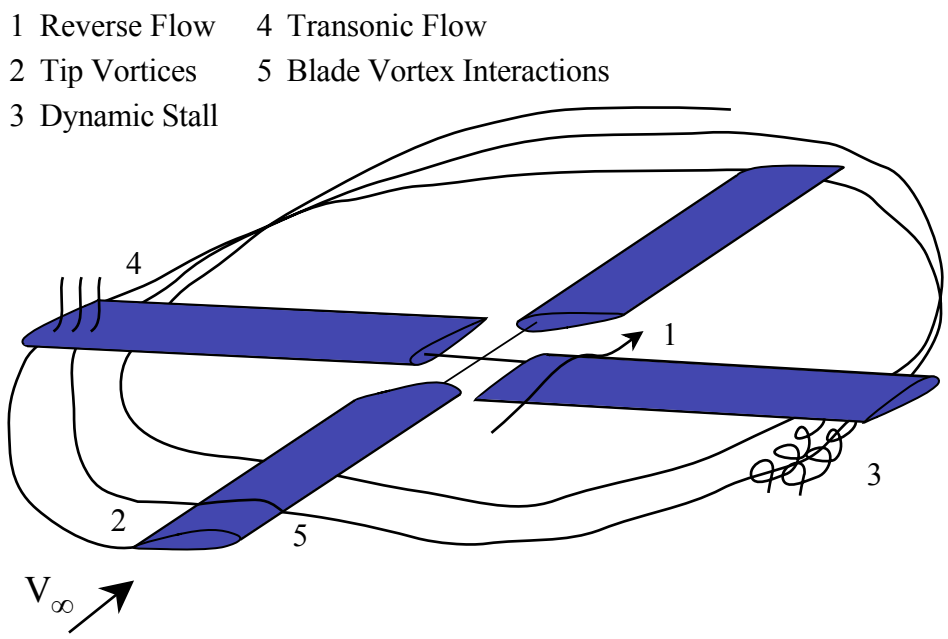


Figure 1: Helicopter rotor flow field aerodynamic phenomena

rotor structural dynamics, wake geometry, and aerodynamic loads. Many comprehensive codes have been written which specifically address rotor predictions. However, these codes concentrate primarily on predicting the structural dynamics of the system and use simplified airloads and inflow methods. CFD methods based on Navier-Stokes equations, coupled with comprehensive codes have improved accuracy, but they are time consuming, even when run in parallel, owing to the required density of the grid to capture flow structures of interest in the CFD simulation. Current focus has thus been on developing more efficient methods for rotor airload predictions that still yield reasonably accurate solutions.

1.2 Previous Work

The first codes that were developed to address the problem of rotor airload predictions are known as the first-generation codes. These codes, although termed comprehensive, are characterized by simplified analyses that are only applicable to specific cases. An overview

of many of these codes can be found in [5]. The second-generation codes were then developed to address many of the problems associated with the first-generation codes. These newer codes have much wider applicability to a range of geometries and incorporate more accurate structural and aerodynamic computations into the analysis. These computations include nonlinear finite element analysis routines as well as unsteady aerodynamic modules and possibly free wake inflow models. Where these codes differ is mainly in the types of finite elements available, the method used to compute the unsteady aerodynamics, and the inflow computations available. Some of the major codes that have been developed as part of this generation include UMARC [6], CAMRAD II [7, 8], RDYNE [9], HOST [10], and DYMORE [11]. UMARC, developed at the University of Maryland, uses finite element analysis to solve the equations of motion of the rotor blades. The finite elements used assume moderate deflections. It also incorporates different aerodynamic and inflow models, as well as being capable of trim analysis. The trim model used solves six force balance equations for a target setting. HOST, developed through the combined effort of ONERA, Eurocopter, and DLR, incorporates the features of the S80 and R53 first-generation codes, and includes expanded features for trim, aerodynamic, and inflow analysis. The trim analysis is based on computations from a harmonic representation of movement and internal state. RDYNE, a code from Sikorsky, incorporates multibody dynamics to solve the rotor equations of motion. To trim, it uses a minimum variance controller on the blade pitch motion, and uses the hub loads as the target values. CAMRAD II, developed by Johnson Aeronautics is a multibody dynamics code that uses finite element analysis along the blades. It uses geometrically exact nonlinear finite elements and assumes moderate deflections. It incorporates an internal free wake and dynamic inflow models, simplified aerodynamics, and a trim model. The current research effort utilizes DYMORE, a code developed at the Georgia Institute of Technology. This multibody code uses geometrically exact finite elements, does not make assumptions about the size of the deflections, and uses an autopilot

trimmer. While this code has not been specifically developed for rotor analysis, its generalized structure allows it to be easily adapted to the task. All of these codes include both structural and aerodynamic computations, but they were developed primarily as structural codes and do not address the aerodynamic aspects of the problem thoroughly. The internal aerodynamic computations provided in these codes are lifting line models and do not take advantage of the accuracy of CFD computations.

While the accuracy of the comprehensive codes has been enhanced by improving the aerodynamic computations, many CFD codes have also been improved by incorporating structural computations. These codes, while primarily intended to model the flow field, have been modified to allow for simplified grid motion during the computations. This grid motion comes in the form of adding either 6DoF models, or prescribed motion models. RANS codes such as OVERFLOW [12, 13], and FUN3D [1, 14, 15, 16, 17] have basic rotor prediction capabilities internally incorporated through the use of overset grid capabilities, and simplified rigid motion computations. One example is the addition of a 1D Euler-Bernoulli beam model to TURNS [18]. The biggest difference between these codes is the form of the Navier-Stokes equations they solve, and the type of model discretization they use. All of the CFD codes mentioned above are structured codes with the exception of FUN3D, which uses unstructured grids. In general, these codes are considered to be accurate but inefficient. However, as computers become faster, and as the use of parallel programming becomes wider, the efficiency of these methods will improve.

In order to take advantage of the accuracy of each type of the above mentioned codes, more generalized inputs are being added to allow for the use of data from other external sources. This is the best way to be able to take advantage of the most accurate computations available from both the structural and aerodynamic codes, without redeveloping advanced methods in order to improve the existing codes internally. For instance, the comprehensive codes will have options to input aerodynamic data from a CFD code, and the CFD codes will have the option to include rigid and elastic motion data computed by structural codes.

The current state of the art to improve rotor airload predictions is to isolate the simplified internal modules and replace them with more advanced codes through external coupling.

In order to evaluate the accuracy of more efficient methods, another type of coupling of interest is the use of advanced wake models coupled with simplified 2D aerodynamic models. These wake models compute the inflow distribution on the rotor disk and pass this data to the structural code where the internal aerodynamic models are used to compute the airloads. Some of the cases which have been tried include coupling with RDYNE and FREEWAKE [19], UMARC and MFW [20], and DYMORE and MFW [21], of which the last is included as part of the work for this thesis. Not all comprehensive codes need to be coupled with an external free wake code, for instance CAMRAD II has its own internal free wake model [22], the effectiveness of which was compared against other free wake couplings [23]. Although this method is more efficient it is not as accurate as coupling with higher fidelity aerodynamic models.

The degree of accuracy and expense of the high fidelity coupling procedures has increased over time. When the earliest code couplings were developed, the component codes, while an improvement over the internal modules, still were not the most accurate available. One of the first published methods was a coupling between CAMRAD and FDR [24], where FDR is a transonic small disturbance code. Later, more accurate full potential codes were tried to compute the aerodynamic loads. Some of these coupled methods include CAMRAD and FPR [25, 26], RDYNE, FPR, and FREEWAKE [27], 2GCHAS and FPX [28], TECH-01 and FPR [29]. As computational expense became less of an issue in the 1990's, coupling between Euler codes and RANS codes with structural computations began to appear. Some of these include coupling Euler computations with nonlinear elastic blade analysis [30], Euler and RANS computations with CSD [31], DYMORE and OVERFLOW [32], DYMORE and an Euler code [33].

In the last part of this decade, interest in this type of CFD-CSD coupling increased and more of these combined high fidelity RANS models began to appear. Some of these are

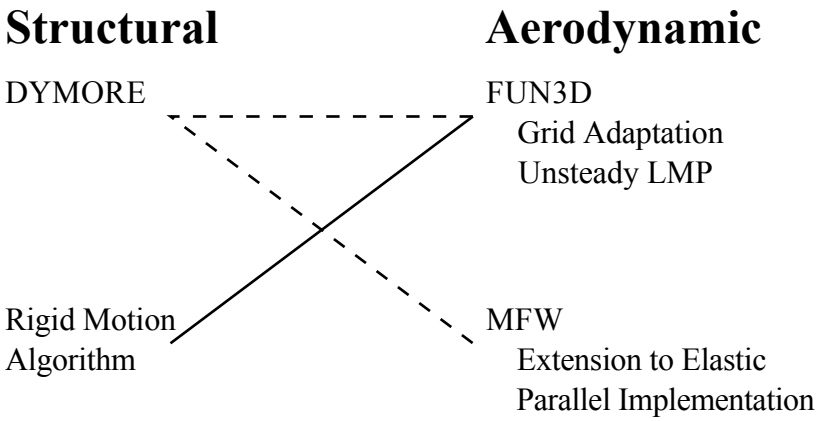


Figure 2: Different codes examined. Dashed lines indicate an external coupling, solid lines indicate internal models

HOST and Waves [34] as well as elsA [35], CAMRAD and OVERFLOW [36], UMARC and TURNS [37], RCAS and OVERFLOW [38, 39], and DYMORE and FUN3D [40], and DYMORE and OVERFLOW. Of these, the coupling of DYMORE and FUN3D is developed as part of this research with comparisons to DYMORE and OVERFLOW courtesy of reference [3]. Many of these methods are discussed in a recent review paper [41]. However, while these CFD-CSD methods are the most accurate methods currently available, they are still time intensive, and thus best used for detailed design analysis. However, for initial calculations, the use of a lower fidelity method is still computationally more efficient for engineering applications. These combined methods show ever improving results, and as these methods, both CFD and free wake, are made even more efficient, their use as design tools will become more widespread.

1.3 Objectives

The objectives of this research are to develop and evaluate coupled methods that improve the underlying physics-based models of the individual codes. The individual models each make certain assumptions about the physics of the situation that impact the accuracy of

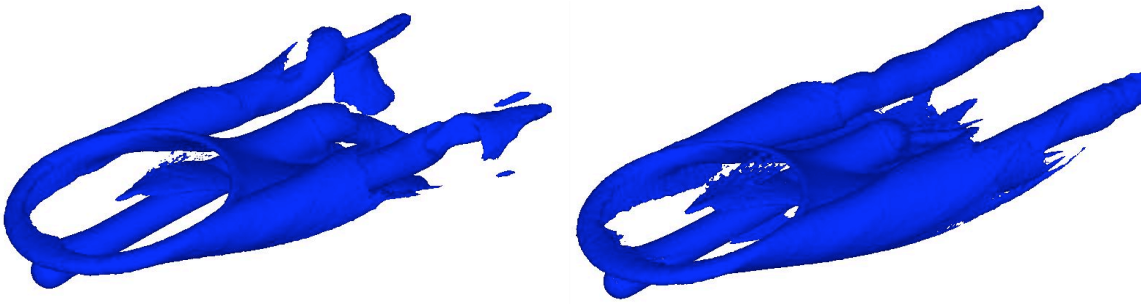
the results. These assumptions are different for each model, where the structural model will assume simplified aerodynamics and the aerodynamics models will assume simplified structural responses. These assumptions change the physics of the computational system being simulated and thus effectively create a different system. Identifying the simplifications made by each computational model will allow for the identification of enhancements that will generate computational systems which are closer to reality.

The scope of the enhancements will include combinations of structural, aerodynamic, and inflow coupling methods, not excluding evaluation of the type of data passed between the codes. The different codes used in this research include a comprehensive code, which is able to model the elastic blade deflections, a CFD code to model the flow field surrounding the rotor, and a free wake code to model the geometry of the wake. Each of these codes has a single strength for which the assumptions of the physical model are reduced. However, these codes are limited by the large number of assumptions made in the other coupled system phenomena. CSD codes are able to model the elastic and rigid structural component elastic deformations accurately, but they use highly simplified aerodynamics, which adds a simplified external load to the system. Thus the structural model, though accurate, is reacting to an approximate flow field, which is not accurate. CFD codes are able to model the flow field well, but without accurate knowledge of the geometry and motion of the physical system, these predictions are not based on the real case, but on a simplified rigid model. Free wake codes are able to model the wake geometry far out into the flow field, but by assuming rigid, unarticulated blade motions this wake is attached to blades which are not moving or deforming like the physical system, and thus the geometry is reacting to a simplified system instead of the real system. The assumptions made by the simplified modules in each code can be reduced by utilizing the data from the more accurate codes using different coupling methods. This will more accurately represent the real system since each code will rely on its strengths and the coupling will reduce the weaknesses. Within these coupled methods the CSD code computes the accurate blade motion and deformation

data and passes this to the aerodynamic codes and the CFD and free wake codes provide accurate flow field predictions to the structural code. As shown in Figure 2, a wide range of possibilities will be considered in a comparison of computational expense versus accuracy, starting with the most computationally intensive and most accurate combination being the CFD-CSD coupling.

1.4 Code Selection

Each of the codes selected for this research are chosen for specific reasons. DYMORE is selected as the CSD code for this research because it uses geometrically exact finite elements and thus does not make any assumptions regarding the size of the finite element deflections. It also is a flexible code, which does not impose restrictions on the kind of rotor model used, because it contains an extensive element library that can be utilized to combine many different mechanical rotor parts. Finally, it has internal aerodynamic models, which are useful when performing both free wake and CFD couplings. FUN3D is selected as the CFD code because it uses unstructured meshes, which allows more control over the grid density distribution than a structured mesh. These grids may be constructed as components for use in overset computations. The use of overset grids reduces the computational expense of moving and deforming the entire grid at each time step. This CFD code is also able to solve both the compressible and incompressible RANS equations, which is useful for rotor applications where some rotor cases may operate either partially or entirely below the compressible boundary. The present code has a steady LMP method implemented into the code. An additional advantage to using an unstructured code is ease with which grid adaptation may be applied. These two features will be briefly demonstrated in the following subsections. These cases apply a steady actuator disk to model the rotor on the GT wind tunnel case. This test case is described in section 4.2 and the steady LMP and the grid adaptation details are explained in sections 3.3.2 and 2.3.2 respectively.

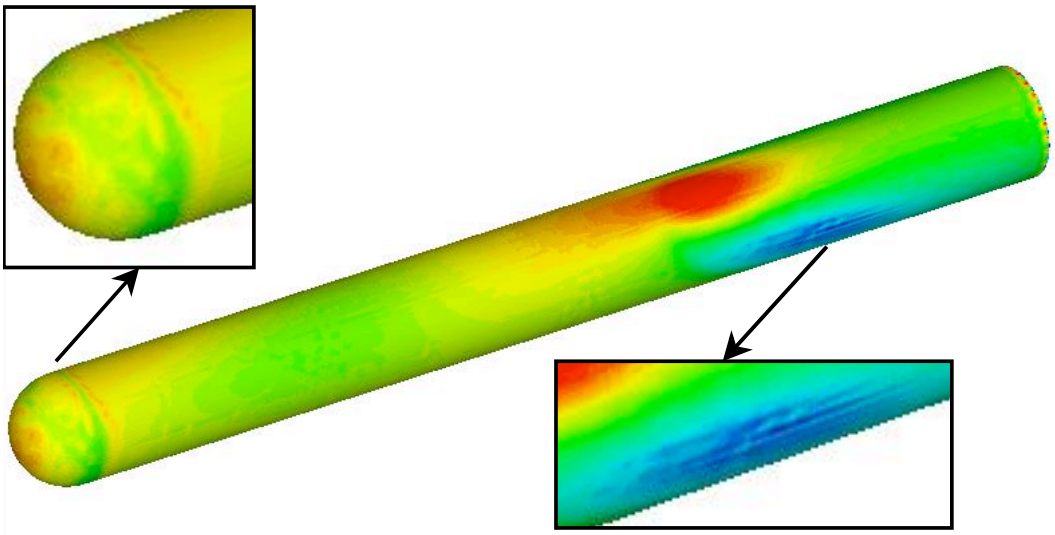


(a) Un-Preconditioned **(b) Preconditioned**

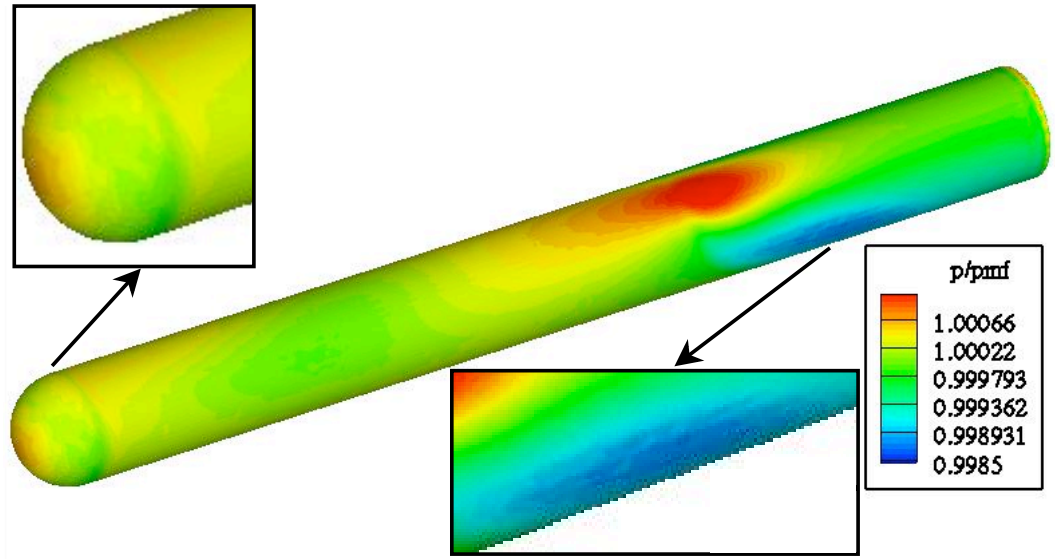
Figure 3: GT actuator disk case iso-vorticity plots of magnitude 0.2 comparing both the un-preconditioned and the preconditioned results

1.4.1 GT Steady Low Mach Number Preconditioning

The steady low Mach number preconditioning method is effective in smoothing out non-physical numerical fluctuations. This simulation was run using the GT actuator disk case using the steady-state compressible Euler equations with and without preconditioning. For the case run with preconditioning the parameter β is set to 0.14. Results comparing the preconditioned and un-preconditioned results are given in Figures 3 and 4. Since this is a steady-state actuator disk case these results correspond to a time-averaged flow over the entire rotor revolution, and do not model the individual time dependent features such as wake vortex propagation. The rotor wake for this actuator disk case, in Figure 3, wraps around the actuator disk as it develops, forming a sheet that trails the rotor disk. This is not the physical result since a real rotor wake is formed by the shedding of vortices from the rotating blades creating an approximately helical stricture which trails the rotor. However, if this wake is averaged, a sheet will be produced because the vortices pass through every point of the sheet at some time during the rotor revolution. The addition of preconditioning has not changed the geometry of the wake sheet, but it has reduced the amount of dissipation seen in the far field results behind the fuselage. In theory this sheet should propagate much further before it dissipates, but CFD computations typically prematurely dissipate this wake much sooner than they should at low forward flight and hover speeds due to the



(a) Un-Preconditioned



(b) Preconditioned

Figure 4: GT actuator disk case non-dimensionalized pressure surface plots comparing both the un-preconditioned and the preconditioned results

high number of iterations with numerical dissipation necessary to reach convergence. The comparison of the surface pressure plots in Figure 4 also predict a more physical result with the addition of preconditioning. The surface pressures for the un-preconditioned case have clear numerical fluctuations resulting from high equation stiffness in the low speed flow around the nose and in the region behind the rotor disk where the wake interacts with the fuselage, as observed in the non-uniform color change in this region. These regions have been magnified for clarity in the figure. The addition of preconditioning has smoothed out the numerical fluctuations in these regions without changing the character of the results. The character of the results has not changed because the wake sheet predicted for both cases is the same in the region near the fuselage surface. Thus, the addition of preconditioning has not changed the interaction of the wake with the fuselage, and therefore, the location of the pressure peaks on the fuselage does not change. This is not the case for the unsteady results because the interaction of the time-accurate wake with the fuselage does change with the addition of preconditioning as will be discussed later. Therefore, the effectiveness of preconditioning has been proven for steady-state flows. This analysis will be extended to the unsteady time-accurate rotor case, which more closely represents a real configuration.

1.4.2 GT Grid Adaptation

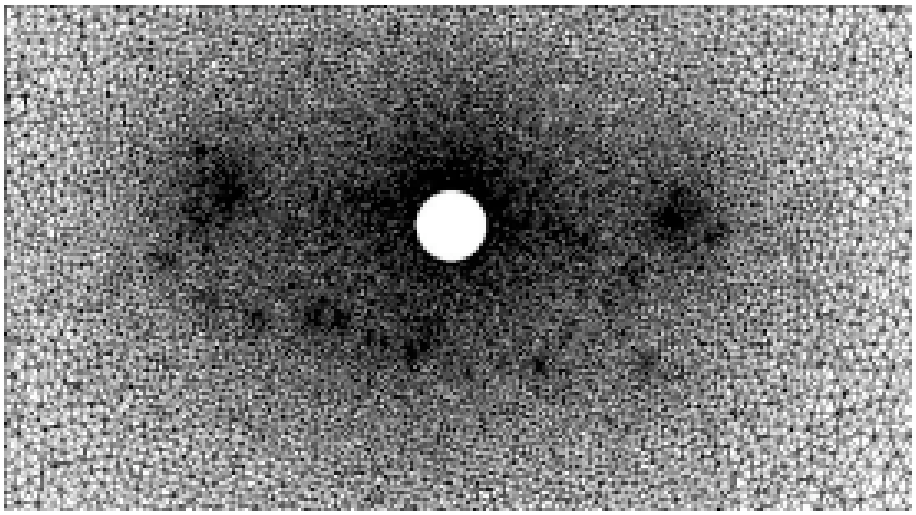
Grid adaptation is a valuable resource in CFD when trying to more accurately resolve the tip vortices that come off of the rotating blades. The main benefit of this adaptation is to not only to obtain a better near body profile of these vortices, but also to reduce their premature dissipation as they propagate into the far field because of the rapid expansion of grid cell size necessary to keep computations within the scope of engineering applications. There, the ability to cluster grids where needed to maintain the wake for a longer distance, while minimizing the number of additional nodes, produces results closer to the actual physics. This case has been run using the steady-state compressible Euler equations. The

adaptation was run with a frozen region around the fuselage measuring 0.015R, with a limit on the maximum cell size of 0.71R. Two adaptation cycles were run for each flow solver restart, and two total adaptation levels were run. All adapted results shown below are for the third restart of the flow solver on the second adapted grid. The original grid has some approximated grid clustering included in the rotor wake region produced during grid generation, thus some refinement is seen in the original grid even though the grid adaptation algorithm has not been applied. The resulting wake geometry for the actuator disk case is in the form of a sheet. The physical comparison of this wake geometry with a time-accurate wake has been described in section 1.4.1.

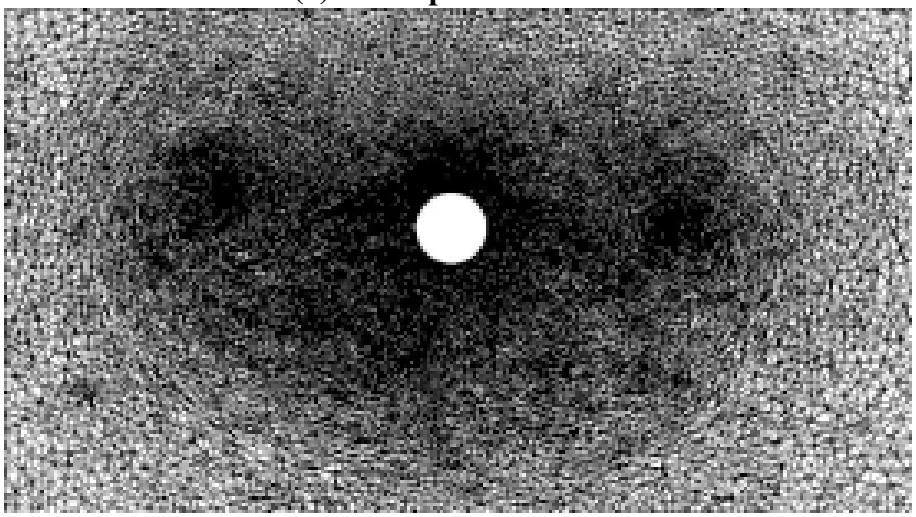
The grid adaptation algorithm is based on using density gradients in the flow to define the refinement locations, and the highest density gradients in the flow field are located within the wake because of the higher velocity gradients in this region; therefore, grid refinement is focused in this region. Figure 5 has one grid slice on the rear of the fuselage at $x = 2.4$ for both the original grid and the adapted cases respectively. Refinement of the wake in the adapted mesh conforms to the corresponding density gradients in Figure 6. Although the original case is able to capture the rotor wake, many of the key features have been smoothed out since the grid does not have enough points in these regions to model rapid spatial fluid changes. The roll-up is not as strong and the sheet connecting the two rolled up regions is not well defined. The addition of grid refinement improves resolution of the wake roll-up and the impact of the wake with the fuselage by adding points in regions where the density is changing the most rapidly. Figure 6b for the refined case illustrates the wrapping of this sheet around the fuselage after it reconnects on the lower side. The three dimensionality of the flow field refinement is illustrated in the comparison of the streamline plots in Figures 7a and 7b. These plots are generated using five vertical streamline rakes of 20 streamlines each located every 0.5R at the inflow boundary of the grid centered at $y = 0$. The consistency of the refinement in the flow direction is illustrated by these figures. The roll-up is much smoother since there are more points in this region with which to model

the wake.

To accommodate the possibility of having to couple different codes together in the future, a proposed standardized coupling format has been implemented, which simplifies the process of switching the codes used in coupling [42]. The usability of this coupling method is addressed by automating as much of the process as possible to reduce both human error and downtime. Chapter 2 will describe each code and the modules implemented within them. Chapter 3 will describe the improvements made to each code and describe the coupling schemes. Chapter 4 will describe the test cases and flight conditions. Chapters 5 - 7 will evaluate the performance of some of the methods. Chapter 8 will compare the results of the cases discussed in previous chapters and discuss the physical meaning of the results. Chapter 9 will provide the conclusions and future recommendations.

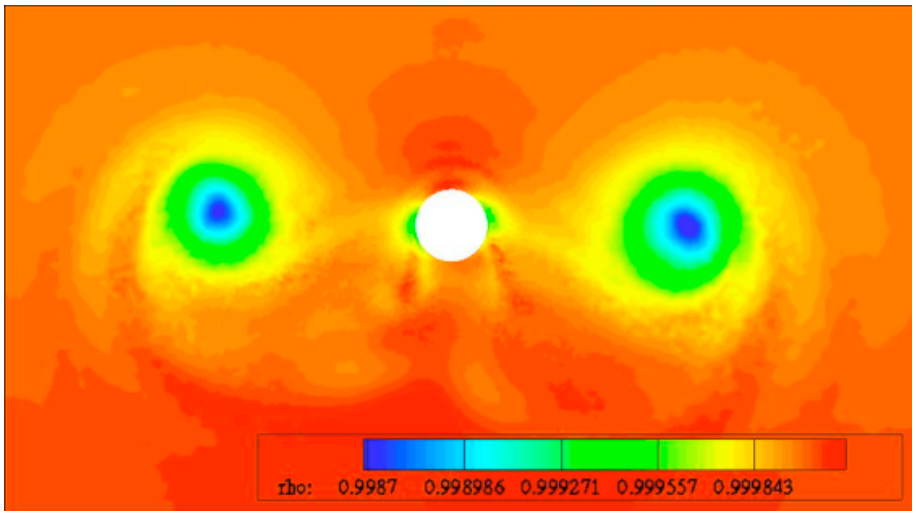


(a) Unadapted Grid

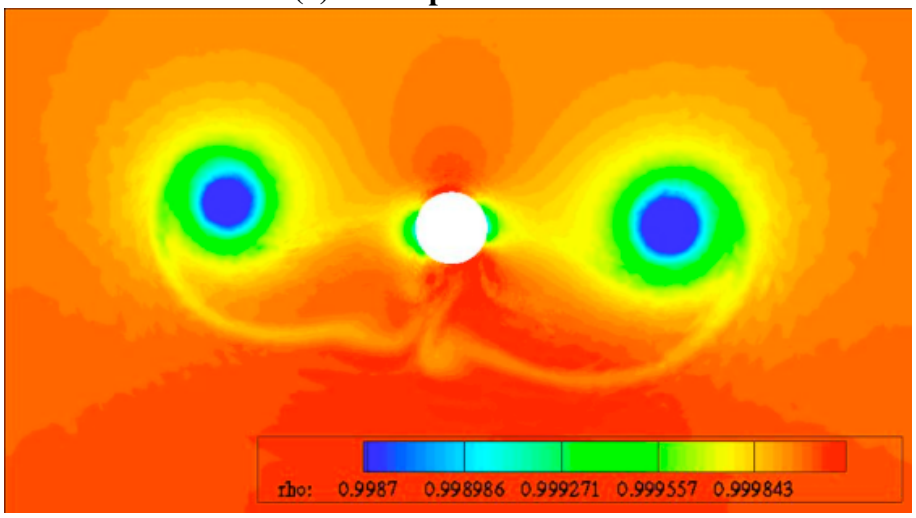


(b) Adapted Grid

Figure 5: GT rear view slice of the wake at x=2.4R for both the unadapted and the adapted grids, comparing the mesh refinement

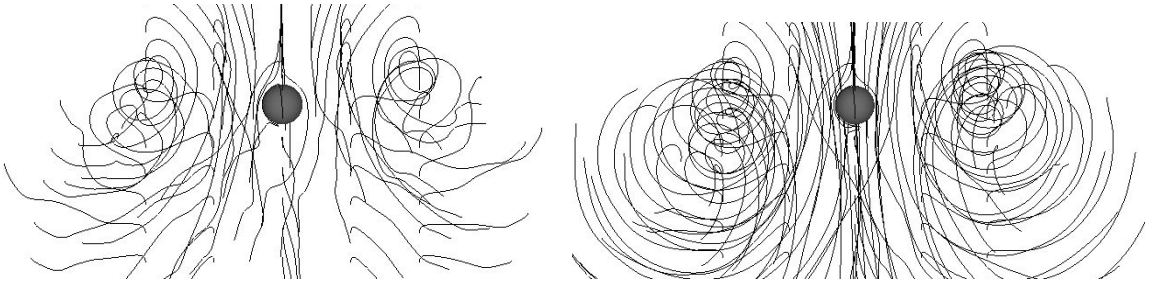


(a) Unadapted Grid



(b) Adapted Grid

Figure 6: GT rear view slice of the wake at x=2.4R for both the unadapted and the adapted grids, comparing the local fluid density



(a) Unadapted Grid **(b) Adapted Grid**

Figure 7: GT front view of the streamlines comparing both the unadapted and the adapted grids

CHAPTER 2

METHODOLOGY

2.1 CSD Model (DYMORE)

The physical phenomena present within complex rotor flow fields must be considered in order to compute the system response reliably. The combination of large displacements, finite rotations, flexible components, and variable material properties make these systems highly nonlinear. The large number of interacting components in the rotor system creates a large number of constraints and degrees of freedom resulting in a sparse linear system. In order to model this type of system, the equations of motion must take these nonlinearities into account. The finite element method used to discretize the problem must be able to handle the high frequency modes that are excited by the numerical round-off error and the tendency of the nonlinearities to transfer energy from the low to the high frequency modes. Many present codes do not take all of these issues into account, making simplifications in the finite element analysis by assuming small or moderate deflections. DYMORE does not make any assumptions with respect to the nature of the motion, thus making this code highly suitable for rotorcraft analysis. The comprehensive nature of this code is defined by the inclusion of computations other than the structural deformation. DYMORE includes simplified 2D unsteady aerodynamics models and a dynamic inflow model to include simplified aerodynamic computations. There is also an auto pilot trimmer, which is used to adjust the blade control angles until target hub loads are predicted.

Overall, DYMORE is a multibody finite element analysis code for arbitrary nonlinear elastic systems [11, 43]. The multibody models are constructed by piecing together basic structural elements, the data for these elements are stored within an element library. Each of these elements has its own known system of equations which when pieced together create

one larger and more complex system of equations. The equation of motion that describes the global system is

$$\tilde{F}_p + \hat{F}_p = \bar{F}_p. \tag{1}$$

This equation represents the balance of the elastic ($\hat{F}$) and inertial ($\tilde{F}$) forces with the external ($\bar{F}$) forces. These individual force components are computed using kinetic and strain energy formulations. The resulting expression after substitution is

$$\frac{d}{dt}(R_{qp}M_{qr}\vec{v}_r) - \dot{u}_r D_{qrp} M_{qr}\vec{v}_r + B_{qp}C_{qr}\epsilon_r = \bar{F}_p. \tag{2}$$

Where u are the degrees of freedom of the system, $\vec{v}$ are the system velocities, and ϵ are the strains, M_{qr} is the mass matrix, C_{qr} is the stiffness matrix, R_{qp} is the rotation matrix, and where intermediate placeholders B and D represent the derivatives

$$B_{pq} = \frac{d\epsilon_p}{du_q} \quad \textit{and} \quad D_{pqr} = \frac{dR_{pq}}{du_r}. \tag{3}$$

The time discretizations of the force components that will be used in the numerical scheme are

$$\tilde{F}^m_p = \frac{R^f_{qp}M_{qr}\vec{v}^f_r - R^i_{qp}M_{qr}\vec{v}^i_r}{\Delta t} - \frac{u^f_r - u^i_r}{\Delta t} D^m_{qrp} \frac{M_{qr}\vec{v}^f_r + M_{qr}\vec{v}^i_r}{2} \tag{4}$$

and

$$\hat{F}^m_p = B^m_{qp}C_{pq}\frac{\epsilon^f_p + \epsilon^i_p}{2}. \tag{5}$$

Where subscripts p, q, and r are summation indices, and m is a time discretization placeholder that represents the time interval t_i to t_f. The scheme used to solve this system of equations is the energy-decaying scheme [43]. This method is unconditionally stable, and unlike the energy-preserving scheme [43], includes high frequency numerical dissipation. The addition of this numerical dissipation is important because of the highly nonlinear nature of rotorcraft problems. These nonlinearities excite the high frequency modes of the system and can interfere with convergence. The discretized equations of motion for the energy-decaying scheme are

$$\tilde{F}^m_p + \hat{F}^g_p = \bar{F}^g_p, \tag{6}$$

and

$$\tilde{F}_p^h - \frac{1}{3}(\hat{F}_p^g - B_{qp}^h C_{qr} \epsilon_r^f) = \bar{F}_p^h, \tag{7}$$

where,

$$\hat{F}_p^g = \frac{\epsilon_q^f - \epsilon_q^j}{u_p^f - u_p^j} C_{pq} \frac{\epsilon_p^f + \epsilon_p^j}{2}, \tag{8}$$

$$\tilde{F}_p^h = \frac{R_{qp}^j M_{qr} \vec{v}_r^j - R_{qp}^i M_{qr} \vec{v}_r^i}{\Delta t} - \frac{u_r^j - u_r^i}{\Delta t} D_{qrp}^m \frac{M_{qr} \vec{v}_r^j + M_{qr} \vec{v}_r^i}{2}, \tag{9}$$

$$\bar{F}_p^g = \frac{1}{2} \int_{-1}^{1} \bar{F}_p d\tau, \tag{10}$$

and

$$\bar{F}_p^h = \frac{1}{2} \int_{-1}^{1} \bar{F}_p \tau d\tau. \tag{11}$$

The superscript g corresponds to a discretization over the time interval t_j to t_f, and superscript h corresponds to a discretization over the time interval t_i to t_j, where $t_i < t_j < t_f$. The final component of the energy-decaying scheme, which balances the system and ensures that energy will not increase over time, is the energy decay statement

$$\Xi^f \leq \Xi^i + \bar{W}, \tag{12}$$

where Ξ is the total mechanical energy, and $\bar{W}$ is the work done by the applied forces. This expression states that the total mechanical energy of the system at the current time step cannot be greater than the energy of the system at the previous time step plus the energy applied to the system over the time interval. The corresponding constraint equations are computed using the Lagrange multiplier method using

$$F_p^{cg} = B_{qp}^g \lambda_q^g \tag{13}$$

and

$$F_p^{ch} = -\frac{1}{3}(F_p^{cg} - B_{qp}^h \lambda_q^j). \tag{14}$$

These expressions turn the system of equations into a combined differential algebraic system of equations. To ensure the exact vanishing of work done by constraint forces

$$\eta_p^f = \eta_p^j = 0, \tag{15}$$

where λ are the Lagrange multipliers, and η are the holonomic constraints.

Model construction in this code is very much like constructing a CFD mesh with vertices, "edges", and boundary conditions. The edges represent the structural elements, both flexible and rigid, as well as the various joints. The vertices represent the connections between these components and can have boundary conditions constraining the translational and rotational motions of the vertices. An example of this model is shown in Figure 8, where the vertices are represented by the dots, and the elements (*i.e.* the "edges") are the lines. The difference between a CFD mesh and this type of model is that the edges do not need to form closed loops in order to form a complete system. In this code the "edge" components are stored in an internal system library that stores the known equations of motion for each component, which are then used to construct the global system of equations. Thus the equations of motion for a complex system may be obtained in a general fashion. This library contains both rigid and elastic components where the elastic components are modeled using geometrically exact finite elements. The library also contains different types of joints, as well as being able to model composite elements [44, 45, 46]. The broad flexibility of the code allows for the incorporation of different types of analysis through substitution of the internal computations with external coupling methods, as will be demonstrated in research.

2.1.1 Unsteady Aerodynamics Modules

DYMORE includes several simplified aerodynamics models to approximately predict the state of the rotor flow field. There are two internal models based on 2D unsteady aerodynamic theory, the Peters model [47], and the Leishman-Beddoes model [48]. Both of these models have implementations of empirical dynamic stall models [49, 50], but these have

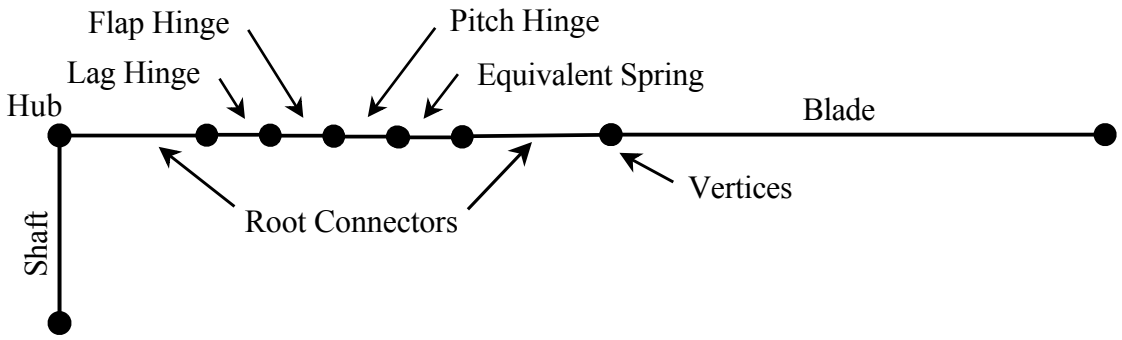


Figure 8: Example DYMORE rotor blade model

not yet been added to the code. There is a dynamic inflow model, which can be used to better predict the rotor inflow distribution. This simplified module is not as accurate as a free wake model, but it does provide an averaged rotor inflow distribution. There is also one external method available to read data computed by a CFD or free wake code. The 2D models within the original codes are approximate and are not able to handle all of the possible physical phenomena associated with rotor aerodynamics, but the option to include external airloads enables higher fidelity aerodynamic models to be used with DYMORE; these models are able to capture additional physical features. The internal methods will be discussed here; development of the external coupling is part of the present research and will be discussed in the next chapter.

2.1.1.1 Peters Unsteady Aerodynamics

One of the available aerodynamics modules in DYMORE is the Peters model [47, 51]. This model is based on two-dimensional unsteady aerodynamic theory, and assumes inviscid, incompressible flow. The resulting expressions for lift, drag, and pitching moment are

$$L = \rho a_0 v w + \rho \frac{b^2}{2} a_0 \dot{w} + \rho \frac{b^2}{2} a_0 v \dot{\alpha}, \tag{16}$$

$$D = \rho b a_0 w V, \tag{17}$$

and

$$M_{c/4} = -\rho\frac{b^4}{16}a_0\ddot{\alpha} - \frac{b}{2}(\rho\frac{b^2}{2}a_0\dot{w} + \rho\frac{b^2}{2}a_0 v\dot{\alpha}). \tag{18}$$

While this model is not empirical in nature it may be modified to take advantage of steady wind tunnel measurements. To account for the measured lift, the expression $c_l = a_0 \sin(\alpha)$ is used to substitute a_0 in the original equations. To account for the measured pitching moment it must first be stated that the theoretical value for the steady pitching moment is zero. Thus any measured non-zero value may simply be added to the unsteady pitching moment using the expression $M^s_{c/4} = 2\rho b^2 c_m V^2$. The same argument may be made for the drag computations. Since the viscous drag is predicted to be zero, the measured value is simply added onto the present value. This measured value is defined along the local wind axis and thus must be decomposed into the airfoil frame components. This rotation will affect both the lift and the drag computations. The final expressions that take into account the availability of wind tunnel data are

$$L = \rho b c_l V v + \rho\frac{b^2}{2}a_0\dot{w} + \rho\frac{b^2}{2}a_0 v\dot{\alpha} + \rho b c_d V w, \tag{19}$$

$$D = \rho b c_l w V - \rho b c_d V v + \rho b c_{d0}(\sqrt{u^2 + v^2} - v)v, \tag{20}$$

and

$$M_{c/4} = 2\rho b^2 c_m V^2 - \rho\frac{b^4}{16}a_0\ddot{\alpha} - \frac{b}{2}(\rho\frac{b^2}{2}a_0\dot{w} + \rho\frac{b^2}{2}a_0 v\dot{\alpha}). \tag{21}$$

These final expressions also have the option to include a tip loss factor that is directly applied to the airload computations. This factor multiplies the steady and unsteady terms, but not the viscous drag terms. Dynamic stall may be accounted for through use of an empirical dynamic stall model [49].

2.1.1.2 Leishman-Beddoes Semi-Empirical Model

The Lieshman-Beddoes model is based on linearized unsteady two-dimensional theory developed using the indicial response formulation [48]. The indicial formulation indicates

that the aerodynamic response can be modeled by a series of impulsive loads that are linearly superimposed on one another to form the total response through Duhamel's integral. In this method, the aerodynamic loads are separated into the circulatory and non-circulatory components. Each component is solved for independently before it is added to the total aerodynamic loads. The expressions used to find the load components are developed as general functions that require the knowledge of the behavior of the specific airfoil being used to compute the coefficients. Thus the method can be considered fully empirical. In actuality, the method is only semi-empirical because the coefficients may be recomputed for different airfoils, and thus this method is suitable for more than one case. The equations for this method are described below, first for the attached flow, then for the separated flow. The attached flow expressions are found by solving the following system of differential equations,

$$\dot{\vec{x}} = \tilde{A}\vec{x} + \tilde{B}\vec{u} \quad \textit{and} \quad \vec{y} = \tilde{C}\vec{x} + \tilde{D}\vec{u}. \tag{22}$$

Where $\vec{u} = \begin{bmatrix} \alpha & \frac{\dot{\alpha}c}{V} \end{bmatrix}^T$, $\vec{y} = \begin{bmatrix} C_N^p & C_M \end{bmatrix}^T$, and $\vec{x}$ is the state array with components x_i, $i = 1, 2, ..., 8$. The matrices used in these systems are,

$$\tilde{A} = -\begin{bmatrix} \frac{V\beta^2 b_1}{b} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{V\beta^2 b_2}{b} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{T_{N\alpha}} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{T_{Nq}} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{b_3 T_{M\alpha}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{b_4 T_{M\alpha}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{V\beta^2 b_5}{b} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{T_{Mq}} \end{bmatrix}, \quad \tilde{B} = \begin{bmatrix} 1 & 0.5 \\ 1 & 0.5 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}, \tag{23}$$

$$\tilde{C} = \begin{bmatrix} \frac{V}{b}a_0\beta^2 A_1 b_1 & (0.25 - x_{ac})\frac{V}{b}a_0\beta^2 A_1 b_1 \\ \frac{V}{b}a_0\beta^2 A_2 b_2 & (0.25 - x_{ac})\frac{V}{b}a_0\beta^2 A_2 b_2 \\ \frac{-4}{MT_{N\alpha}} & 0 \\ \frac{-1}{MT_{Nq}} & 0 \\ 0 & \frac{A_3}{Mb_3 T_{M\alpha}} \\ 0 & \frac{A_4}{Mb_4 T_{M\alpha}} \\ 0 & \frac{-\pi A_5 b_5 \beta V}{8b} \\ 0 & \frac{7}{12MT_{Mq}} \end{bmatrix}^T, \text{and} \quad \tilde{D} = \frac{1}{M}\begin{bmatrix} 4 & 1 \\ -(A_3 + A_4) & \frac{-7}{12} \end{bmatrix}. \quad (24)$$

Where the coefficients A_i and b_i are the indicial response coefficients determined from the unsteady test data. After the normal and moment coefficients have been found the corresponding span and drag forces may be computed using

$$C_c = a_0\alpha_{eff}^2 \quad \text{and} \quad C_D = C_N^p \sin(\alpha) - C_c \cos(\alpha). \quad (25)$$

Where $\alpha = atan(w/v)$ and $\alpha_{eff} = \beta^2(V/b)(A_1 b_1 x_1 + A_2 b_2 x_2)$. The time constants are

$$T_{N\alpha} = \frac{k_{N\alpha}}{(1 - M) + \pi\beta M^2(A_1 b_1 + A_2 b_2)}\frac{2b}{a}, \quad (26)$$

$$T_{Nq} = \frac{k_{Nq}}{0.5(1 - M) + 2\pi\beta M^2(A_1 b_1 + A_2 b_2)}\frac{2b}{a}, \quad (27)$$

$$T_{M\alpha} = \frac{k_{M\alpha}(A_3 b_4 + A_4 b_3)}{b_3 b_4(1 - M)}\frac{2b}{a}, \quad (28)$$

and

$$T_{Mq} = \frac{7k_{Mq}}{15(1 - M) + 3\pi\beta M^2 A_5 b_5}\frac{2b}{a}. \quad (29)$$

Where $k_{N\alpha}$, k_{Nq}, $k_{M\alpha}$, and k_{Mq} are the time constant multipliers, which may be determined from the unsteady test data.

The detached flow components of the aerodynamic loads are found by solving

$$C_N^f = a_0 \left(\frac{1 + \sqrt{f''}}{2} \right)^2 \alpha_{eff} + \frac{1}{M}(4\dot{x}_3 + \dot{x}_4), \tag{30}$$

$$\begin{aligned} C_M =& (k_0 + k_1(1 - f'') + k_2 \sin(\pi f''^m)) a_0 \frac{V}{b} \beta^2 (A_1 b_1 x_1 + A_2 b_2 x_2) + \\ & \left(\frac{-\pi A_5 b_5 \beta V}{8b} \right) x_7 + \frac{-1}{M}(A_3 \dot{x}_5 + A_4 \dot{x}_6) + \frac{-7}{12M} \dot{x}_8 + c_{m0}, \end{aligned} \tag{31}$$

$$C_c^f = \eta a_0 \alpha_{eff}^2 \sqrt{f''}, \tag{32}$$

and

$$C_D^f = C_N^f \sin(\alpha) - C_c^f \cos(\alpha). \tag{33}$$

Where $f'' = f' - D_f$,

$$D_{fn} = D_{f(n-1)} exp\left(\frac{\Delta S}{T_f} \right) + (f_n' - f_{n-1}') exp\left(\frac{\Delta S}{2T_f} \right), \tag{34}$$

and

$$f' = \begin{cases} 1 - 0.3exp(\frac{\alpha_f - \alpha_1}{S_1}) & \alpha_f \leq \alpha_1 \\ 0.04 + 0.66exp(\frac{\alpha_1 - \alpha_f}{S_2}) & \alpha_f > \alpha_1 \end{cases}. \tag{35}$$

The trailing edge separation point may be found approximately using Kirchoff-Helmholtz theory and measured airfoil static load data

$$f = \left[2\sqrt{\frac{C_N}{a_0 \alpha}} - 1 \right]^2. \tag{36}$$

The constants α_1, S_1, and S_2 are found by fitting equation f' on curve f. The parameters k_0, k_1, k_2, and m are found by fitting the equation

$$\frac{C_M}{C_N} = k_0 + k_1(1 - f) + k_2 \sin(\pi f^m) \tag{37}$$

to the experimental data for $\frac{C_M}{C_N}$. The time constant T_f is found by using the unsteady test data. This model also includes computations to account for dynamic stall [50], which will not be included here.

2.1.1.3 Dynamic Inflow Module

The Peters-He dynamic inflow model [52] is incorporated internally into DYMORE. It is a state-space method based on 3D unsteady acceleration potential theory, and neglects wake distortion and vortex roll-up. The consequence of these assumptions is that this method is similar to a prescribed wake analysis and thus misses some flow discontinuities. This method also has slow convergence. The main purpose of this method is to provide design capability, not to perform a detailed loading analysis.

The theory is based on linearized incompressible potential flow theory, generated from the conservation of mass and momentum. The effect of blade bending out of plane on the wake is not included, loading is normal to the rotor plane, and the components of the inflow velocity, other than in the normal direction, are neglected. The resulting induced inflow expression in the time domain is

$$E^{-1}[\bar{w}]^{*} + L^{-1}[\bar{w}] = \Phi^{A} + \Phi^{V} = \Phi. \tag{38}$$

Where the invertible linear matrix operators $L[\ ]$ and $E[\ ]$ relate the pressure coefficients to the quasi-steady inflow and inflow acceleration respectively, and $\bar{w}$ is the normal component of the induced velocity. The pressure, Φ, is divided into, convective, Φ^{V}, and unsteady, Φ^{A} terms. The operators are related to these pressures by

$$\bar{w}^{*} = E[\Phi^{A}] = -\frac{\partial \Phi^{A}}{\partial z}\Big|_{\eta=0}, \tag{39}$$

and

$$\bar{w} = L[\Phi^{V}] = -\frac{1}{V_{\infty}} \int_{0}^{\infty} \frac{\partial \Phi^{V}}{\partial z} d\xi. \tag{40}$$

These are time domain expressions, where the superscript $(.)^{*}$ denotes the derivative with respect to non-dimensional time.

In reality, the lift drops off at the tip of the blade. However, a large number of terms are needed in the inflow model in order to model this phenomenon correctly. In order to account for the loss of lift at the tip of the blade using a smaller number of series expansions,

a tip-loss factor is included. This factor is based on a modified form of Prandtl's correction factor and is defined as

$$f = \tanh \frac{1 - \frac{s}{s_t}}{1 - \lambda}. \tag{41}$$

Where s is the curvilinear radial coordinate of the airstation, s_t is the curvilinear radial coordinate of the tip of the blade, and λ is the tip loss factor.

2.1.2 Auto Pilot Trimmer

Trim analysis is accomplished using DYMORE's internal auto pilot trim analysis [53]. The auto pilot trim is performed in three steps. First, during the initialization, DYMORE is run using a set of prescribed pitch settings until the loads converge. Next, three perturbation runs are performed in which the first three harmonics of pitch are independently varied using

$$\hat{u}_j = u_{j_{ref}} + \Delta_j \qquad j = 1, 2, 3. \tag{42}$$

Where for each update, the harmonics of the blade pitch settings are changed by Δ_j, and the simulation is run to convergence. The gradual application of changing the control outputs during the perturbation steps is performed using

$$\hat{u}_i(t_i) = \begin{cases} \frac{\Delta_i[1 - cos(\frac{\pi t_i}{T})]}{2} & \text{if } t_i < T \\ \Delta_i & \text{if } t_i \geq T \end{cases}. \tag{43}$$

Where t_i is the initial start time of the perturbation evaluation, and T is the time constant. The results from these perturbation and reference runs are then used to compute a Jacobian matrix. This is a 3-by-3 matrix:

$$[J] = \begin{bmatrix} \frac{dy_1}{d\hat{u}_1} & \frac{dy_2}{d\hat{u}_1} & \frac{dy_3}{d\hat{u}_1} \\ \frac{dy_1}{d\hat{u}_2} & \frac{dy_2}{d\hat{u}_2} & \frac{dy_3}{d\hat{u}_2} \\ \frac{dy_1}{d\hat{u}_3} & \frac{dy_2}{d\hat{u}_3} & \frac{dy_3}{d\hat{u}_3} \end{bmatrix} \tag{44}$$

in which each entry is the derivative of an input (y) with respect to an output ($\hat{u}$). Thus, for each input (thrust, hub rolling moment, and hub pitching moment) there is a relation with each output (three coefficients of the blade pitch Fourier series). These relations allow DYMORE to compute the magnitude of the step change needed for the blade pitching so that the loads move gradually toward the target hub loads. This computation is performed as follows,

$$\vec{\hat{u}}_f = \vec{\hat{u}}_i + \Delta t J^{-1} G(\vec{y}_o - \vec{y}_m). \tag{45}$$

Where $\vec{\hat{u}}_f$ are the outputs at the final point of the time step, $\vec{\hat{u}}_i$ are the outputs at the initial point of the time step, $\vec{y}_m$ are the present sensor measurements, $\vec{y}_o$ are the target values, Δt is the time step size, and J and G are the Jacobian and gain matrices, respectively. This computation is repeated at each time step, and the simulation is considered converged when the three harmonics of blade pitch stop changing.

Extending the application of this trimmer to the coupling scheme requires no changes. For all steps in the trim analysis (reference, three perturbation, and simulation steps) DYMORE acquires data from the external codes using the same method that would be used if trim analysis were not applied. Through the passing of the structural data to the external aerodynamic codes the trim settings are implicitly included in the aerodynamic analysis.

2.2 Free Wake Model (MFW)

The FSI coupling strategy allows for a "plug and play" approach to include different types of analysis into DYMORE. One goal of this project is to allow incorporation of any free wake code into any CSD code, using DYMORE and MFW as a testbed for this analysis. MFW [54, 55] computes the rotor inflow in a time-accurate sense using discrete line vortex filaments in a Lagrangian scheme using Weissinger-L analysis [56] to compute the circulation distribution. The computations are then performed by applying the Navier-Stokes equations in the velocity-vorticity form

$$\frac{\partial \vec{\omega}}{\partial t} = -(\vec{V} \cdot \vec{\nabla})\vec{\omega} + (\vec{\omega} \cdot \vec{\nabla})\vec{V} + \nu \vec{\Delta} \cdot \vec{\omega}. \tag{46}$$

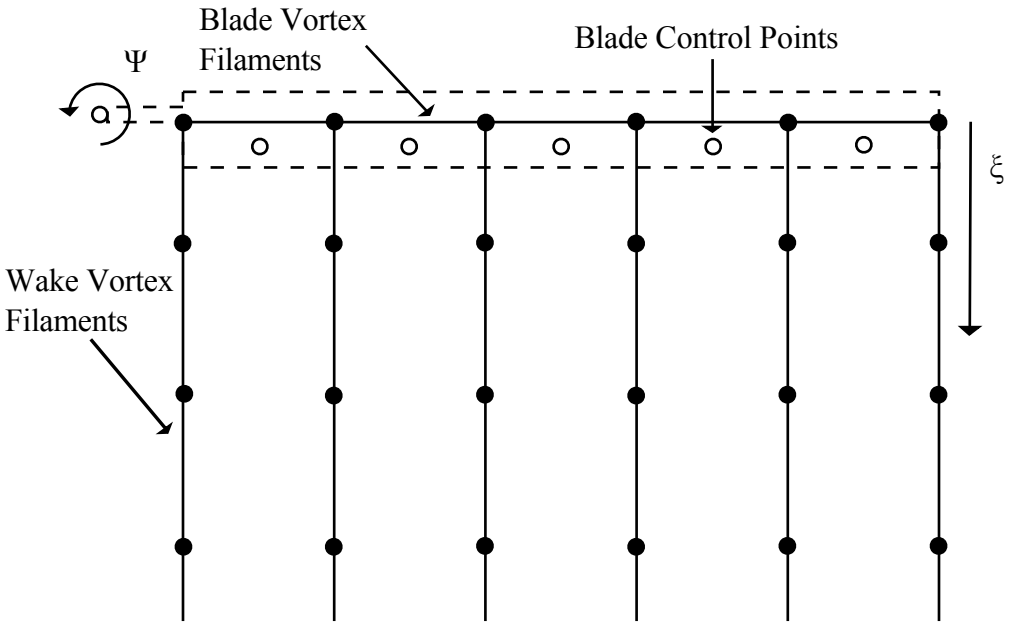


Figure 9: Model discretization in MFW

Where the right hand side terms represent convection, strain, and diffusion, respectively. This expression is simplified for rotor analysis by also assuming inviscid, irrotational conditions and constant vortex strength line segments. When following a particular wake marker the equation becomes

$$\frac{\partial \vec{r}(\psi,\xi)}{\partial \psi} + \frac{\partial \vec{r}(\psi,\xi)}{\partial \xi} = \frac{1}{\Omega}\vec{V}(\vec{r}(\psi,\xi)). \tag{47}$$

This expression is in the form of the 1D wave equation. The solution of this wave equation is relatively simple. However, most of the time spent during the free wake computations is spent computing the velocity term on the RHS.

Models in the free wake code are generated by discretizing the blade and wake into vortex filaments, as is illustrated in Figure 9. The number of wake trailers and blade segments may be varied in number and position, depending on the problem being solved. In this mesh, the two variables ξ and ψ represent the spatial variation and the temporal variation of the model respectively. It is this variable selection that reduces the equations into the form

of the 1D wave equation. There are two different discretization schemes, relaxation and time-accurate, that are then applied to this equation, each representing a different solution methodology. Of these two methods, the first developed is the relaxation scheme [57]. In this method the temporal and spatial derivatives are discretized using a five-point difference scheme. This method assumes periodicity of the solution. However, the periodicity assumption is not suitable for coupling, it is the time-accurate form of the code that enables the coupling with external comprehensive codes and is therefore the one used in the present research. The time-accurate method [58] does not assume periodicity and is discretized by applying a five-point central difference scheme to the spatial derivatives, and a predictor-corrector scheme with second order backward differencing to the temporal derivative. In this method the predictor step takes the form

$$\tilde{r}_{\xi\psi} = r_{\xi-\Delta\xi\psi-\Delta\psi} + \frac{2}{\Omega}\left(\frac{\Delta\psi\Delta\xi}{\Delta\psi+\Delta\xi}\right)V, \tag{48}$$

where

$$V = V_\infty + \tilde{V}_{ind\xi-\Delta\xi\psi-2\Delta\psi} + V_{ext}\left(r_{\xi\psi}\right), \tag{49}$$

$r_{\xi\psi}$ is the wake vertex position at (ξ, ψ), and $\tilde{r}_{\xi\psi}$ is the intermediate wake vertex position. The corrector step is computed using

$$\begin{aligned} r_{\xi\psi} = r_{\xi-\Delta\xi\psi-\Delta\psi} + \tfrac{2}{\Omega}\left(\tfrac{\Delta\psi\Delta\xi}{\Delta\psi+\Delta\xi}\right)V + \tfrac{1}{16}\Big(r_{\xi\psi} - 3r_{\xi\psi-\Delta\psi} + \dots \\ +3r_{\xi\psi-2\Delta\psi} - r_{\xi\psi-3\Delta\psi} + r_{\xi-\Delta\xi\psi} - 3r_{\xi-\Delta\xi\psi-\Delta\psi} + 3r_{\xi-\Delta\xi\psi-2\Delta\psi} - r_{\xi-\Delta\xi\psi-3\Delta\psi}\Big), \end{aligned} \tag{50}$$

where

$$V = V_\infty + \frac{1}{4}\left(\tilde{V}_{ind\xi-\Delta\xi\psi-\Delta\psi} + \tilde{V}_{ind\xi\psi-\Delta\psi} + V_{ind\xi-\Delta\xi\psi} + V_{ind\xi\psi}\right) + V_{ext}\left(\tilde{r}_{\xi\psi}\right). \tag{51}$$

The velocity expressions shown are decomposed into freestream, induced, and external components, of which the induced term is the most time intensive to compute. To compute the induced velocity, the influence of each vortex filament on the each blade segment is found through application of the Biot-Savart law. The Biot-Savart law computes the velocity induced by a single vortex of length l and constant strength Γ at a given point in space

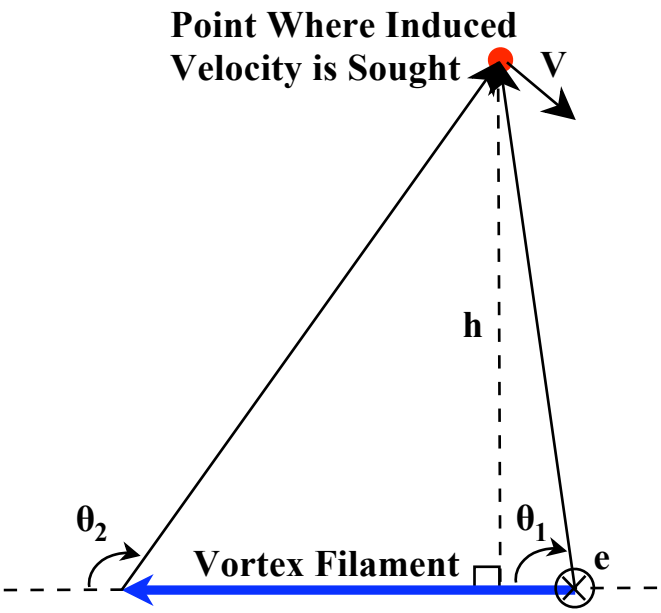


Figure 10: Variables associated with the computation of the discrete Biot-Savart law

defined by $\vec{r}$, and is of the form

$$\vec{V} = \frac{\Gamma}{4\pi} \int_l \frac{\vec{dl} \times \vec{r}}{\left|\vec{r}\right|^3}. \tag{52}$$

For a straight discrete vortex filament this expression becomes

$$\vec{V} = \frac{\Gamma}{4\pi h} \left(\cos\theta_1 - \cos\theta_2\right) \vec{e}, \tag{53}$$

where the meaning of each variable is illustrated in Figure 10. In this expression $\vec{e}$ is the unit vector cross product of the vector $\vec{l}$ along the vortex filament and the vector $\vec{r}$ from the end point of the vortex filament to the given point in space where the induced velocity is computed. The total induced velocity at each blade segment control point is then found by summing the velocity contribution of all of the vortex filaments in the mesh.

The strengths of the vortex filaments are computed by using the Weissinger-L blade lift distribution. The bound circulation along the blade is computed using the influence coefficient matrices that relate the induced velocity of the bound and the near-wake vortices to the blade control points. The resulting linear system

$$\left[ICM_b + ICM_{nw}\right] \Gamma_b = V_\infty \left(\alpha_{eff}\right) \tag{54}$$

is solved for the bound circulation (Γ_b). The structural computations assume rigid motion and only allow pitch and flap motion. The flap displacement is computed by solving the simplified flapping equations of motion for a rigid blade,

$$\ddot{\beta} + \nu_\beta^2 \beta = \frac{M_\beta}{I_\beta \Omega^2} + \frac{\omega_o^2}{\Omega^2}\beta_p. \tag{55}$$

The pitch motion is then computed by implementing a rotor wind tunnel trim procedure. This method trims to the rotor thrust and the first harmonic of the flap motion. The resulting linear system

$$[\Delta C_T \ \beta_{1c} \ \beta_{1s}]^T = J\,[\theta_0 \ \theta_{1c} \ \theta_{1s}]^T = 0 \tag{56}$$

is solved for the pitch motion needed to trim the rotor. The aerodynamic loads used to compute the rotor thrust used in the trim computations are computed using unsteady theory based on linear subsonic flow using the indicial method [59]. The assumptions of rigid motion and partial articulation limit the applicability of this code to elastic CSD coupling, which assumes neither rigid motion nor partial articulation. Modifications to the model, as well as to the computations, are necessary to allow for elastic blade motion. The expansion of this code to enable the possibility of coupling with CSD codes will be discussed in the next chapter.

2.3 CFD Model (FUN3D)

The CFD code evaluated for this project is FUN3D [1, 14, 15, 16, 17]. This code solves either the compressible or the incompressible URANS equations using unstructured grids. The unstructured grid is discretized into 3D tetrahedral elements, and the numerical scheme used to solve these equations is a node-centered finite-volume scheme. This code has a parallel implementation, with a supporting code (PartyMPI) to partition the grid to the desired number of processors. The general equations of motion to be solved are

$$\frac{\partial (QV)}{\partial t} + \int_{\partial V} (\vec{F} - \vec{F}_v) \cdot \hat{n} dS = 0. \tag{57}$$

Where the conservative variables are defined by

$$q = \begin{bmatrix} \rho & \rho u & \rho v & \rho w & E \end{bmatrix}^T, \tag{58}$$

and the primitive variables are

$$q' = \begin{bmatrix} \rho & u & v & w & p \end{bmatrix}^T, \tag{59}$$

where Q is a volume average of q, and the inviscid and viscous flux vectors are given respectively by

$$\vec{F} = \begin{bmatrix} \rho(u - W_x) \\ \rho u(u - W_x) + p \\ \rho v(u - W_x) \\ \rho w(u - W_x) \\ (E + p)(u - W_x) + W_x p \end{bmatrix} \hat{i} + \begin{bmatrix} \rho(v - W_y) \\ \rho u(v - W_y) \\ \rho v(v - W_y) + p \\ \rho w(v - W_y) \\ (E + p)(v - W_y) + W_y p \end{bmatrix} \hat{j} + \begin{bmatrix} \rho(w - W_z) \\ \rho u(w - W_z) \\ \rho v(w - W_z) \\ \rho w(w - W_z) + p \\ (E + p)(w - W_z) + W_z p \end{bmatrix} \hat{k}$$

and

$$\vec{F}_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{zx} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{zx} - q_x \end{bmatrix} \hat{i} + \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{zy} \\ u\tau_{xy} + v\tau_{yy} + w\tau_{zy} - q_y \end{bmatrix} \hat{j} + \begin{bmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\tau_{xz} + v\tau_{xz} + w\tau_{zz} - q_z \end{bmatrix} \hat{k}.$$

Where (u, v, w) are the flow velocity components, and (W_x, W_y, W_z) are the grid velocity components. The stress tensors are computed using Stokes' hypothesis as

$$\tau_{xx} = \frac{2}{3}\frac{M_\infty}{Re}(\mu + \mu_t)\left(2\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y} - \frac{\partial w}{\partial z}\right), \tag{60}$$

$$\tau_{yy} = \frac{2}{3}\frac{M_\infty}{Re}(\mu + \mu_t)\left(2\frac{\partial v}{\partial y} - \frac{\partial u}{\partial x} - \frac{\partial w}{\partial z}\right), \tag{61}$$

$$\tau_{zz} = \frac{2}{3}\frac{M_\infty}{Re}(\mu + \mu_t)\left(2\frac{\partial w}{\partial z} - \frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}\right), \tag{62}$$

$$\tau_{xy} = \tau_{yx} = (\mu + \mu_t)\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right), \tag{63}$$

$$\tau_{xz} = \tau_{zx} = (\mu + \mu_t)\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right), \tag{64}$$

and

$$\tau_{yz} = \tau_{zy} = (\mu + \mu_t)\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right). \tag{65}$$

The heat flux vectors are given by

$$q_x = -\frac{M_\infty}{Re(\gamma - 1)}\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial T}{\partial x}, \tag{66}$$

$$q_y = -\frac{M_\infty}{Re(\gamma - 1)}\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial T}{\partial y}, \tag{67}$$

and

$$q_z = -\frac{M_\infty}{Re(\gamma - 1)}\left(\frac{\mu}{Pr} + \frac{\mu_t}{Pr_t}\right)\frac{\partial T}{\partial z}. \tag{68}$$

Where μ_t is the eddy viscosity, Re is the Reynold's number, and Pr is the Prandtl number. The perfect gas equation of state is

$$p = (\gamma - 1)\left(E - \rho\frac{u^2 + v^2 + w^2}{2}\right). \tag{69}$$

The viscosity is found using Sutherland's law as

$$\mu = \frac{1 + C^*}{T + C^*}T^{3/2}, \tag{70}$$

where $C^* = 198.6/T_\infty$. The spatial discretization uses a dual mesh formulation in which the control volumes are defined as the region surrounding each node. A 2D example of this type of mesh is given in Figure 11, where the blue dots indicate the original mesh and the red dots indicate where the dual mesh nodes are located. The control volume for the center node of the original mesh is indicated by the shaded region. The spatial discretization for the inviscid terms is

$$\int_{\partial V} \vec{F} \cdot \vec{n} dl \approx \sum_{i=1}^{N} \Phi\left(q^+, q^-; \hat{n}\right) \times l_i \tag{71}$$

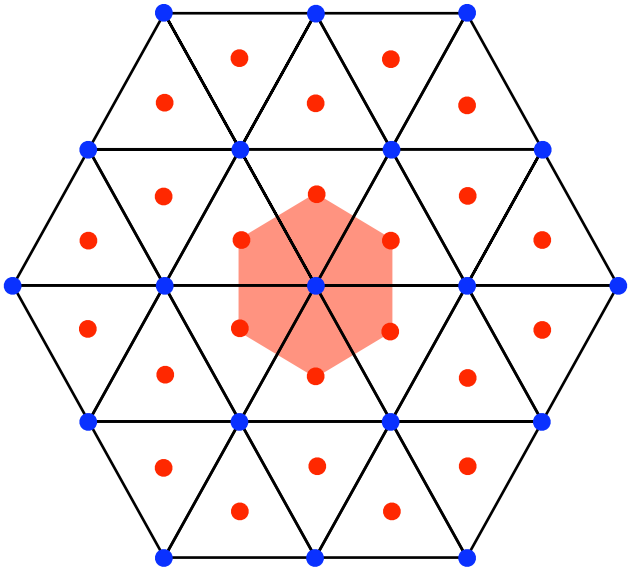


Figure 11: Unstructured dual mesh example for 2D grid segment

where N is the number of edges in the dual mesh surrounding the node of interest, q^+ represents the solution on the left hand side, q^- represents the solution on the right hand side, $\hat{n}$ is the average directed area pointing outward from the control volume boundary, and $\vec{F}$ is the inviscid flux vector defined above. The numerical flux Φ is formed using Roe's approximate Riemann solver as

$$\Phi\left(q^+, q^-; \hat{n}\right) = \frac{1}{2}\left(\vec{F}(q^+; \hat{n}) + \vec{F}(q^-; \hat{n})\right) - \frac{1}{2}\left|A(\hat{q}; \hat{n})\right|\left(q^+ - q^-\right). \tag{72}$$

Where $|A(\hat{q}; \hat{n})|$ is a matrix containing the variables on the cell face. For higher order accuracy, the solution q at each node is extrapolated to the boundaries of the dual mesh control volume using

$$q_{face} = q_{node} + \psi \nabla q \cdot \vec{r} \tag{73}$$

where $\vec{r}$ is the vector from the node to the center of the edge, and $0 < \psi < 1$ is computed using a flux-limiter such as the min-mod or Venkatakrishnan limiters. The gradient ∇q is

found by solving the following linear system,

$$\begin{bmatrix} \Delta x_1 & \Delta y_1 & \Delta z_1 \\ \Delta x_2 & \Delta y_2 & \Delta z_2 \\ \vdots & \vdots & \vdots \\ \Delta x_N & \Delta y_N & \Delta z_N \end{bmatrix} \nabla q = \begin{Bmatrix} q_1 - q_0 \\ q_2 - q_0 \\ \vdots \\ q_N - q_0 \end{Bmatrix}. \tag{74}$$

This linear system is solved using the Gram-Schmidt method. This reduces the sensitivity to stretched grids, which negatively impacts the accuracy of the solution.

The viscous components are solved in a similar manner to the inviscid components. The viscous terms take the form

$$\int_{\partial V} \left[(\mu + \mu_t) \frac{\partial u}{\partial x} \right] \cdot \frac{\partial \hat{n}}{\partial x} dl. \tag{75}$$

These terms are solved using a finite volume method by first computing the quantity $[\mu + \mu_t]$ as an average of the surrounding nodes. The kinematic viscosity μ is found from Sutherland's law, and the eddy viscosity μ_t is found by using a turbulence model such as the Spalart-Allmaras [60] or Menter's SST [61] models. A linear distribution of the solution is assumed across each face, making the gradients constant. These gradients are then found using Green's theorem. Assuming that the control volumes used above do not deform, the temporal discretization can be formulated by representing the governing equations in the form

$$V \frac{\partial Q}{\partial t} = R. \tag{76}$$

A series expansion using backward differencing applied to this expression, with a pseudo-time term to reduce linearization errors, yields

$$V \frac{Q^{m+1} - Q^m}{\Delta \tau} + \frac{V}{\Delta t} (\phi_{n+1} Q^{m+1} + \phi_n Q^n + \phi_{n-1} Q^{n-1} + \phi_{n-2} Q^{n-2} + \dots) = R^{m+1}. \tag{77}$$

Where the coefficients ϕ determine the time-accuracy of the equation. Table 1 lists possible sets of coefficients for different orders of accuracy. The first and second order schemes are stable for all time steps; the third order scheme is not. The $BDF2_{opt}$ scheme is a linear

combination of the second and third order schemes and is stable for all time steps. This expression may be further manipulated to yield

$$\left[\left(\frac{V}{\Delta\tau}+\frac{V\phi_{n+1}}{\Delta t}\right)I-\frac{\partial R^m}{\partial Q}\right]\Delta Q^m = R^m - \frac{V\phi_{n+1}}{\Delta t}(Q^m-Q^n)-\frac{V\phi_{n-1}}{\Delta t}(Q^{n-1}-Q^n)- \frac{V\phi_{n-2}}{\Delta t}(Q^{n-2}-Q^n)-\ldots. \tag{78}$$

Where the steady-state scheme may be obtained by setting $\Delta t = \infty$. Point Gauss-Seidel sweeps are used at each time step to solve this linear system. The time step size is held constant across the entire mesh. However, local time stepping in which $\Delta\tau$ is not held constant across the mesh is used between time steps to advance the solution in pseudo-time.

Table 1: FUN3D temporal discretization coefficients, reference [1]

order	ϕ_{n+1}	ϕ_n	ϕ_{n-1}	ϕ_{n-2}
1^{st}	1	-1	0	0
2^{nd}	$3/2$	-2	$1/2$	0
3^{rd}	$11/6$	-3	$3/2$	$-1/3$
$BDF2_{opt}$	$3/2-\phi_{n-2}$	$-2+3\phi_{n-2}$	$1/2-3\phi_{n-2}$	$-0.58/3$

2.3.1 Overset Grid Computations

The overset analysis capabilities are included through integration of the DiRTLib and SUGGAR codes into FUN3D. DiRTLib and SUGGAR were developed by Dr. Ralph Noack [62, 63]. The implementation of these codes into FUN3D is described in reference [2]. Of these two codes the overset grid creation is performed by SUGGAR. This code combines the component grids by first reading in the background grid, which contains the outer freestream boundary, and the component grids, whose nodes lie entirely within the background grid. These grids are then transformed according to user specified commands given in the input file. Next, the node classification is performed as shown in Figure 12. The overlapping points from the background grid, shown in black, that lie within the solid surfaces of the component grid are classified as hole points. The remaining points are then

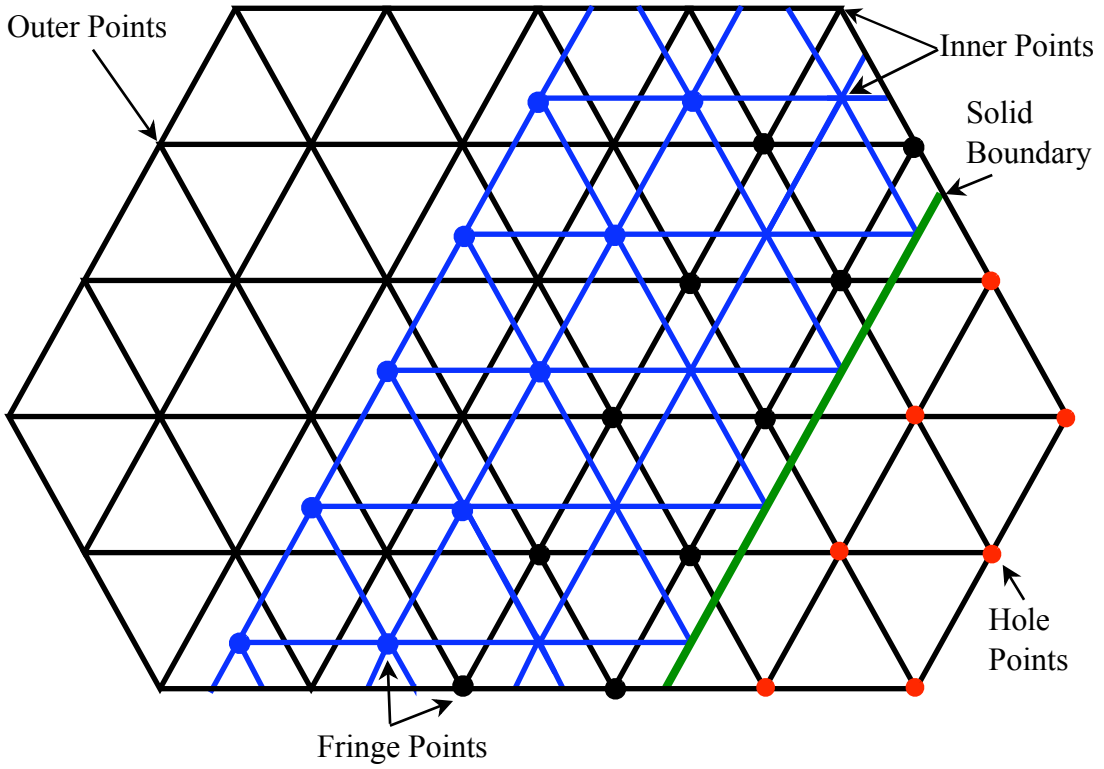


Figure 12: 2D example of overset grid node classifications

classified as either inner, outer, or fringe points. The inner points are all points that lie in the overlapped grid region between the hole and the outer boundary of the component grid, shown in blue. The outer points lie in the region of the background grid that does not overlap. The fringe points lie on the boundary between the outer and inner points, and the inner and the hole points. These are the points at which information is passed between the two grids through interpolation. The orphan points are points for which SUGGAR is unable to create an interpolation stencil. Data at these orphan points are passed by interpolating, not between background and component grid, but entirely within the grid in which the point is located. This situation impacts the accuracy of the solution, and should be avoided when possible by matching the component and background grid resolutions more closely. The node classifications are then written to a DCI file. For moving body cases a DCI file is written for every possible grid configuration used in a simulation.

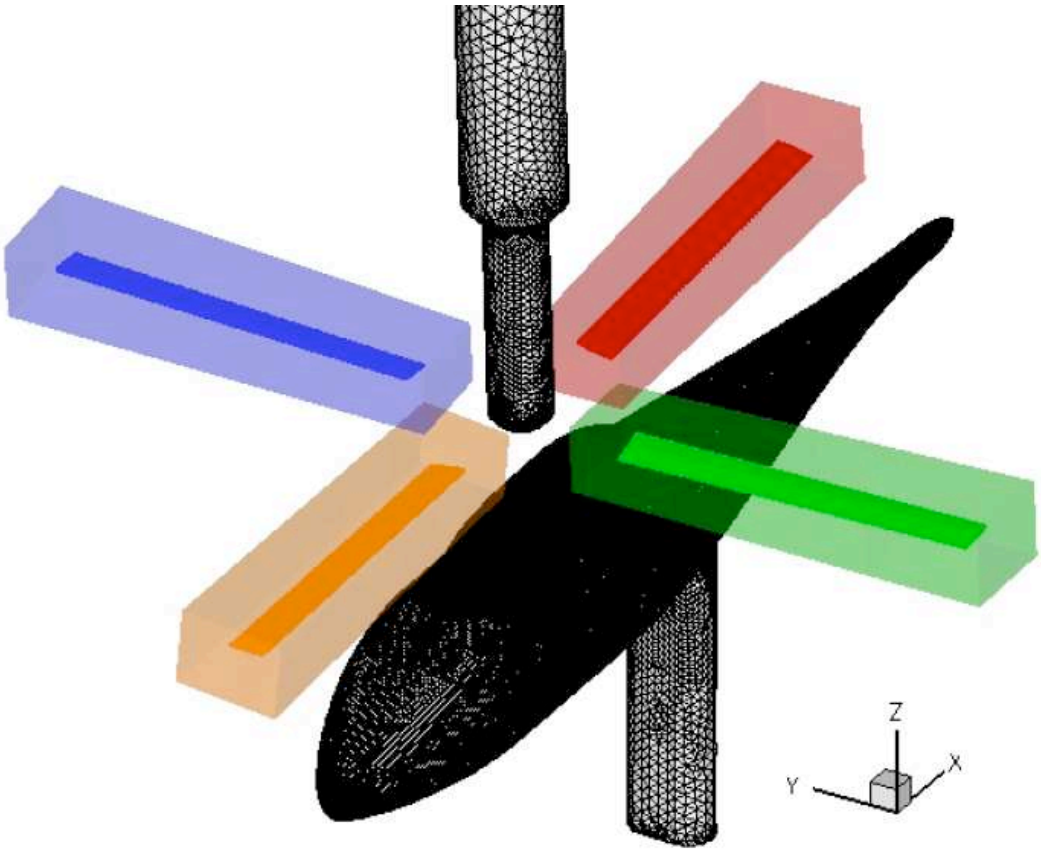


Figure 13: Overset Robin helicopter configuration used in FUN3D, image from reference [2]

The data in the DCI file is read into FUN3D, and is incorporated into the analysis through calls to DiRTLib functions. The purpose of DiRTLib is to provide an interface between SUGGAR's overset computations and a CFD code. The functions in this library use the information output by SUGGAR to identify which nodes to use to perform the interpolation between component grids at every time step. For moving body simulations a different DCI file is read at each time step. In the present research, the resulting model used in FUN3D consists of a combination of a fuselage (background) grid and individual blade (component) grids, as shown in Figure 13. These are moving body cases and thus DCI files are needed for every time step over one rotor revolution. This type of overset capability is advantageous because the motion is applied to the blade grids without disturbing the

fuselage grid. This method is less computationally intensive than if the entire system were modeled as one grid, where the entire system would then need to be modified at each step.

2.3.2 Grid Adaptation

The ability to refine the rotor wake using grid adaptation is a useful tool when predicting helicopter loads. Refinement of the tip vortices, not only near the rotor, but also in the far field, will reduce the premature dissipation of the rotor wake into the far field as well as better predicting the vortex profiles in the near field. FUN3D utilizes more than one type of grid adaptation. The most common feature-based adaptation method, which is found in most other CFD codes, will be discussed here. The feature-based grid adaptation method [64, 65] modifies the grid based on density gradients. First, a converged flow solution is obtained, from this solution the edge jump parameter $J_e = |\Delta\rho|$ is computed. This parameter is used to compute the local error adaptation intensity

$$\hat{I} = \max_{incident\ edges} \left(\frac{J_e}{J_t} \right), \tag{79}$$

where J_t is the user defined jump tolerance. From this intensity parameter the characteristic mesh size h_1 is computed as

$$h_1 = h^0 \min \left(C, \left(\frac{1}{\hat{I}} \right)^{\omega} \right). \tag{80}$$

Where ω is a relaxation parameter, C is a coarsening factor that limits the maximum size of the cells allowed in smooth regions, and h^0 is the characteristic cell size of the corresponding element in the current grid. The anisotropic metric is then computed through the Mach Hessian matrix,

$$H = \begin{bmatrix} \frac{\partial^2 M}{\partial x^2} & \frac{\partial^2 M}{\partial x \partial y} & \frac{\partial^2 M}{\partial x \partial z} \\ \frac{\partial^2 M}{\partial x \partial y} & \frac{\partial^2 M}{\partial y^2} & \frac{\partial^2 M}{\partial y \partial z} \\ \frac{\partial^2 M}{\partial x \partial z} & \frac{\partial^2 M}{\partial y \partial z} & \frac{\partial^2 M}{\partial z^2} \end{bmatrix}. \tag{81}$$

This matrix can be decomposed into eigenvectors and eigenvalue matrices as $H = R\Lambda R^T$.

The metric tensor M is related to this decomposition through $M = R|\Lambda|R^T$. The eigenvectors relate to the element spacing in the three principle directions specified by the eigenvectors as,

$$|\Lambda| = \begin{bmatrix} |\lambda_1| & & \\ & |\lambda_2| & \\ & & |\lambda_3| \end{bmatrix} = \begin{bmatrix} \left(\frac{1}{h_1}\right)^2 & & \\ & \left(\frac{1}{h_2}\right)^2 & \\ & & \left(\frac{1}{h_3}\right)^2 \end{bmatrix}. \tag{82}$$

The characteristic length h_1, computed using density gradients, is related to the Mach Hessian by scaling the three eigenvectors such that

$$\lambda_1 = \left(\frac{1}{h_1}\right)^2. \tag{83}$$

Where h_1 was computed using equation 80. Physical space is mapped into the transformation space using a Jacobian through the expression

$$x' = Jx. \tag{84}$$

Where x is in physical space and x' is in the transformed space. The Jacobian is related to the metric tensor through

$$M = J^T J. \tag{85}$$

The length of each edge is found through the relation

$$l = \sqrt{x'^T x'}. \tag{86}$$

Substitution to produce an expression in terms of the physical coordinates yields,

$$l = \sqrt{(Jx)^T (Jx)}. \tag{87}$$

Which, through substitution of equation 85, is equivalent to the expression

$$l = \sqrt{x^T M x}. \tag{88}$$

Equation 88 is used to compute the edge lengths to determine which elements should be split or removed. For each node the longest and shortest incident edges are computed using

equation 88. These edges are then compared to a user defined spacing metric. If the longest edge is greater than this metric then the edge is split. If all of the edges are shorter than the spacing metric then the shortest edge is removed. Once the insertions and removals have been applied, face and edge swapping improve the mesh quality. Face and edge swapping maximize the aspect ratio of the mesh elements by adjusting the node connectors. Finally, the existing nodes are shifted to smooth the mesh by improving the cell quality in the mapped space using equation 85. This smoothing is applied such that the elements are anisotropic in physical space. The current methodology does not allow for grid adaptation in the viscous layers, and the ability to refine the inviscid boundary surface using CAD geometry has not been used, since only viscous cases have been investigated. To avoid this region, a frozen cell layer is defined which prevents grid adaptation from being applied near solid surfaces. Therefore, while the surrounding flow field is refined, the boundary layer remains the same. Thus, all results presented will look at the flow field refinement rather than the surface results.

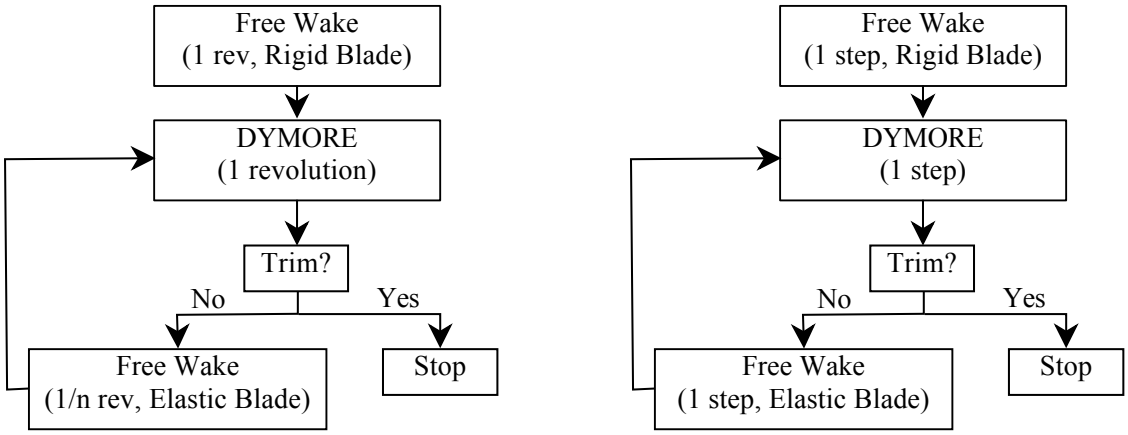
CHAPTER 3

METHODOLOGY MODIFICATIONS

3.1 FSI Coupling Schemes

The data format used for both the CSD-FW coupling, and the CFD-CSD coupling is based on the scheme developed by Nygaard *et al.* [42], extended to include the possibility of coupling with a free wake code, and for delta loads coupling [66]. The basis of this method is the use of file I/O to transfer data between the codes being coupled. This method is chosen for both its generality and its portability. Other methods may implement machine dependent features that may not be available on all systems. This will unnecessarily reduce the number of systems on which the coupling can be run. There is some risk of data loss through this type of file passing that is a function of the precision with which the data are stored in the text files. Using higher precision will reduce the impact of these errors. All that is required to implement this method is a subroutine to read the FSI data files and a subroutine to convert the data from FSI format to the code format. After these subroutines have been implemented, any two codes can be coupled with no modification.

The data passing may be performed manually or may be automated. The CSD-FW coupling is fully automated to demonstrate how this may be accomplished, but the CFD-CSD coupling is performed manually so that the iterations may be analyzed in more detail. Specifically, for the CSD-FW method, each code will have a data file and a flag file. The data file contains the data to be passed to the other code, and the flag file tells the other code when the data are ready to be read. A series of wait loops in each code watch for the generation of these flag files to signal when the data are ready to be read. For the CFD-CSD method the flag files and the wait loops are not implemented, the data files are passed manually to each code as they are generated. This method is not automated because



(a) Loose Coupling **(b) Tight Coupling**

Figure 14: CSD free wake coupling algorithm compared for different file passing intervals

it takes a relatively long time for the CFD computations to finish, and it is not necessary to have the CSD code wait that long, which will require the dedicated use of a node that could otherwise be free. In general, the advantage of this type of coupling is that extensive modifications to the source code, which would be required in order to couple the codes directly through subroutines, are not required. Thus, less effort is expended to include the data from each code into the other, and this leads to the "plug and play" functionality mentioned earlier. While the above description describes the general theory behind the coupling methods, the specific application of this method leaves many variables, including the interval at which the data are passed between the codes and the type of data passed between the codes.

3.1.1 Free Wake and Structural Dynamic Coupling

The variability in the coupling method lies both in the interval at which the data are passed between the codes, as illustrated in Figure 14, and the type of aerodynamic data that is passed to the inflow model. Addressing the first issue of the data passing interval, the flow charts in Figure 14 show the two dominant methods, loose coupling and tight coupling.

While these charts specifically show the CSD-FW method, they also illustrate the two dominant file passing intervals in a general sense. Other passing intervals such as extended-loose coupling, which passes data at more than one revolution, is useful for CFD-CSD coupling, and is described in more detail later as the delta loads coupling method. Another method is semi-tight coupling, which passes data between different model resolutions, but at an interval less than one rotor revolution. This method has been tried and was abandoned because it demonstrated no advantage over the other two methods These two dominant types are considered the most important for computational efficiency and the ability to model maneuvering flight respectively. These methods will be described in more detail in the following section.

3.1.1.1 Data Passing Intervals

In loose coupling, information is passed between the codes for multiple time steps; Figure 14a illustrates this process. Specifically in this research, loose coupling is taken to mean that data are passed every one rotor revolution. The inflow is extracted every $1/n$ revolution, where n is the number of blades in the rotor, and is written out to a standardized text file, with a flag file to signal that the file is ready to be read. Initially, inflow data are computed using the free wake code internal rigid blade analysis. The inflow data are then read by the CSD code, where the inflow is added directly to the local velocity at each radial station. This inflow is used as a constant during the CSD code aerodynamic calculations for one rotor revolution. During this time, the blade deflections, rotations, and effective angle of attack are written out to similar ASCII files, with corresponding flag files. Data are passed in this fashion until a converged solution has been obtained in which the pitch motion has been adjusted until target hub loads have been predicted. The advantage of this type of coupling is primarily to improve the computation time required to obtain the data. Not only is the free wake code run at a fraction of the physical time of the structural code, but the free wake code can also be run at a coarser azimuthal resolution than the structural

code, using interpolation to fill in the gaps when the inflow data are read into DYMORE. The use of extended-loose coupling, which converges the data before it is passed between the codes, has been investigated. This type of coupling did not show any advantage since the same solution was always obtained. This convergence method has been abandoned in favor of the current loose coupling strategy.

For tight coupling, data are passed every time step as shown in Figure 14b. This requires that both the free wake code and the structural code use exactly the same time step size (azimuthal resolution), and that both of the codes run for the same amount of physical time. This eliminates the possibility of running the free wake code at $1/n^{th}$ the time of the structural code. Thus, this type of coupling takes far more time to complete a simulation. However, this additional time is necessary if an unsteady solution is sought, such as for maneuvering flight where the data provided by both of the codes cannot be assumed constant at each revolution. As in the loose coupling analysis, the tight coupling is started from dead start with under-relaxation in the form of a density ramp added to the DYMORE computations. This density ramp is necessary because DYMORE initializes its aerodynamic environment in a vacuum. This initial vacuum leads to very high transient loading for the first 35^o to 45^o of the simulation, which quickly destabilizes the run as the transient data are passed back and forth through the data files. The only time saving device available for this type of coupling is the ability to use the Jacobian matrix computed in the corresponding loose-coupled simulation to jump to the last step of the trim computations. This allows the tight coupling to skip the reference step and the three perturbation steps and directly start the simulation step, which saves computation time. The process used to apply tight coupling does not change regardless of which types of codes are being used. Both CSD-FW and CFD-CSD coupling will pass data at every time step between the codes until convergence has been achieved. Specifically, for the CFD-CSD coupling, the use of a Jacobian from a previous CSD simulation may speed up the subsequent CSD iterations, since the time to compute the airloads using CFD is significantly longer, this time savings

can be significant.

In general, the advantage to loose coupling is the ability to save time, by introducing the possibility of using different model resolutions in the two codes. This allows a relatively computationally expensive code such as the free wake model to use a coarser model, and thus takes less time to run, than the more efficient CSD code. However, this method is only useful if the flight conditions remain constant throughout the run. For maneuvering flight, the data must be passed more frequently, and thus the model resolutions must be the same. This is more computationally expensive, but it is capable of solving a wider range of flight conditions. However, while changing the data passing interval is useful to save computational time when possible, it has absolutely no impact on the final results, since the data passing interval does not change the physical problem being solved or the assumptions applied. The only differences in the results from this coupling are grid dependency issues, which may arise from the use of coarser models in the loose coupling scheme.

3.1.1.2 Variations in Coupling Data Used

In the case of CFD-CSD coupling there are no variations in the data passed between the two codes. The CFD code will always pass the airloads, and the CSD code will always pass the elastic blade deflections. However, in the case of CSD-FW coupling this is not necessarily the case. In this instance, if the method used to pass data between the codes is held constant, the data contained in the files being passed between codes can still be varied. This possibility arises because there is more than one way to compute the circulation inside the free wake code. A few variations have been evaluated for this project, and will be described below.

One method that can be used to pass data between the codes is to pass the effective angle of attack into the inflow model and use this angle of attack to compute the circulations along the blade. The reasoning behind this method is to use the previously implemented

Weissinger-L methodology [56] in MFW, and thus make as few modifications to the existing code as possible. The final form used inside of MFW to compute the circulations results in a linear system is

$$[ICM_b + ICM_{nw}]\,\Gamma_b = V_\infty\left(\alpha_{eff}\right). \tag{89}$$

This system uses the effective angle of attack to compute the blade bound circulation. No other modifications are required for this method other than to substitute the internal effective angle of attack computations in the free wake model with the data from the FSI file.

Another method used to pass data is the frozen strength method. In the inflow model the bound, and thus the near and far wake, vortex strengths are frozen to the initially computed values, while the wake geometry, including the elastic blade deflections, is left free to move. This method can evaluate the effect of wake geometry on the computations independent of the vortex strength computations. This is a good way of evaluating the sources of instability in the code during debugging, but its restrictive assumptions make it the least accurate form of coupling.

A more basic method that is used to pass data is the open-loop method. This method does not pass any aerodynamic loading data between the codes, only inflow and deformation data. This method can also be useful for initial debugging, but is also a more approximate method than the angle of attack coupling. This method allows MFW to use its own aerodynamic computations to compute the circulation and is only influenced by DYMORE through the wake trailer placement boundary conditions along the blade. DYMORE is then only influenced by MFW through the inflow as before.

A fourth possible method is the load passing method. This method has been implemented, but there are stability issues. No configuration has been found which makes this method stable. The reasons for this instability are quite clear. The Peters aerodynamic model in DYMORE currently being used computes a high load boundary between the region of reverse flow and the outer region. This is because the simplified aerodynamic model

is not set up to handle high angles of attack. This reverse flow boundary computes high loads and has low local velocity, which leads to a computation of high circulation, which computes even higher inflow velocity and thus even higher loads. This cycle continues until the loads cause the code to become numerically unstable. Since this region is restricted to a band encircling the region of reverse flow, it would be reasonable to assume that the problem is with the zero velocity band that must, by definition, define the region of reverse flow. However, attempts to account for this phenomenon in the free wake code have been unsuccessful, always resulting in instabilities. This high load boundary is a feature that is unique to the aerodynamic model in DYMORE and is not computed at the location where the velocities are zero, but encircles this region at some distance, and therefore changing or modifying the aerodynamic model to account for this region should make further attempts to use the loads passing method possible.

The theory behind using this loads passing method is as follows: The loads on each section of the blade are passed to the inflow model where the loads are used to compute the bound vortex circulation through the Kutta-Joukowski theorem, which directly relates the circulation to the airloads, and is of the form

$$L = \rho U \Gamma. \tag{90}$$

This method requires more modification to the inflow code than the angle of attack method including the elimination of all instances of the influence coefficient matrices in the computations and eliminating the use of the Weissinger-L computations. The use of the influence coefficient matrices has been found to destabilize the code under all circumstances when using loads passing and has instead been replaced with direct application of the Biot-Savart law. This modification eliminates a catastrophic instability with a more easily identified instability, which is traced to the aerodynamic model. The run time is not significantly increased by eliminating the influence coefficient matrix (ICM), because in order to initially make the code accommodate elastic blade deflections, the computation of the ICM was moved such that it is computed at every time step, which in and of itself increases the run

time of the code by about 20%. Thus replacing the ICM with a direct computation has little effect on present code efficiency.

Out of all four methods considered, the angle of attack coupling method is chosen as the preferred method because it allows both structural and aerodynamic coupling between the models, and its stability does not suffer from the assumptions imposed by the simplified aerodynamics models. The angle of attack that is passed to MFW is accurately represented because it depends only on the local velocity components. By comparison, the load method is limited by the assumptions made in the aerodynamic model, and thus is not accurately represented in critical regions such as the reverse flow region, leading to catastrophic instabilities in the free wake computations. The last two structurally based methods are considered only for comparison since the degree of coupling is significantly lower than for the first two methods.

3.1.2 CFD and Structural Dynamic Coupling

If the types of model formats used in the coupling are similar, then coupling the codes is straight forward, requiring little modification to the data as it is passed. This is the case with the CSD-FW coupling method since both codes use the same lifting-line format to model the rotor blades. However, if the two models are different, then a majority of the time may be spent adapting the data between models. This is the case for CFD-CSD coupling, because the CSD code uses a simplified finite element and lifting-line model to define each blade, and the CFD code uses an unstructured grid to model the entire flow field and blade surface. In order to pass data between the two codes, the structural data, which is sent in lifting-line format, must be applied to the entire unstructured blade surface. The resulting airload data, which is computed across the entire unstructured grid surface, must then be converted into lifting-line format. The complication of this conversion is compounded by the use of an unstructured rather than a structured grid. This difficulty results from the unavailability of mapping the grid to cartesian I, J, K indices, which naturally divide a

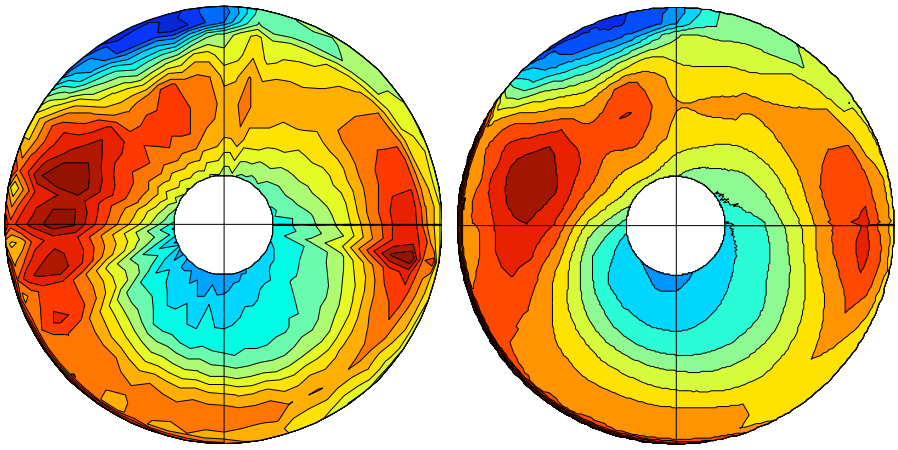
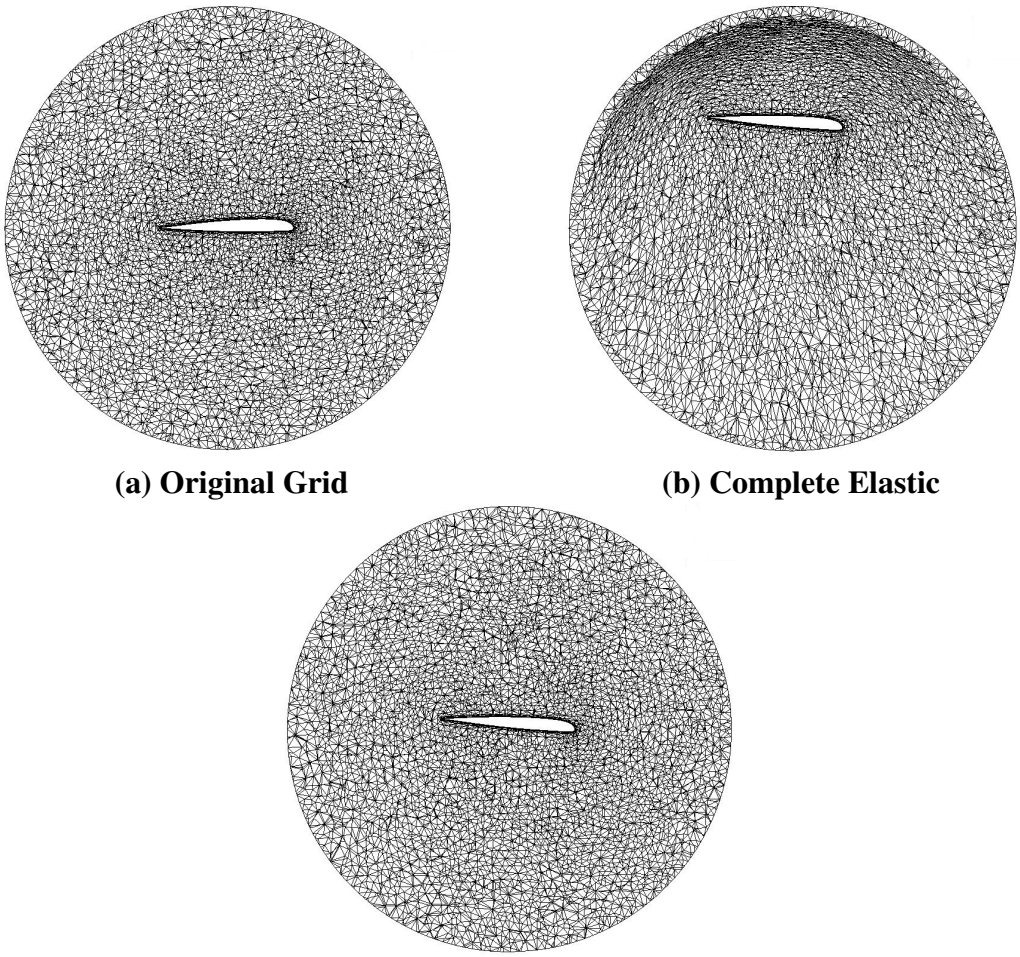


Figure 15: Comparison of the impact of azimuthal structural data interpolation methods on CFD airloads results

structured blade grid into sections and simplify algorithm development.

When converting the structural deformation data, each grid point in the unstructured CFD grid is deformed by first interpolating the lifting-line data to the desired radial and azimuthal location. The use of a higher order cubic spline interpolation has been determined to be better than a lower order linear interpolation since the linear interpolation does not accurately represent the motion of the blades at unspecified locations. This point is illustrated by Figure 15 where the results for linear and cubic spline interpolation are compared. For data provided at every 15^o, using linear interpolation generates jumps in the data every 15^o since only the magnitude of the known data are considered. Cubic spline interpolation smoothes the motion of the blades and more accurately represents the motion at unspecified locations by matching both the magnitude and the slope at the known data locations. After interpolation, each grid point is deformed by applying the translation and rotation indicated by the interpolated data, taking into account the grid point position on the surface relative to the interpolated data point at the quarter chord location. A new volume grid is then computed using the CFD code, which has an internal volume mesh deformation capability.



(a) Original Grid **(b) Complete Elastic**

(c) Mixed Elastic-Rigid

Figure 16: Comparison of different volume deformation methods for the blade cross section at $\frac{r}{R} = 0.7$

This method is an acceptable way of computing the motion, but it has one major drawback, if the background grid used for the overset blade is not large enough, like the one shown in Figure 16a, then the volume grid points will cluster in one place, as seen in Figure 16b. This clustering can interfere with the CFD code by creating sudden jumps in cell size between grid regions and sliver cells, which result in higher computational error. This error is directly related to the truncation errors in the Taylor series expansion of the finite differencing. In particular, consider the first derivative of a function $u(x)$ over two adjoining cells of size Δx and $d_x \Delta x$. The corresponding Taylor expansion of the finite difference

expression is

$$\frac{u_{i+1} - u_{i-1}}{x_{i+1} - x_{i-1}} = \frac{\partial u}{\partial x} + \frac{d_x - 1}{2}\Delta x \frac{\partial u^2}{\partial x^2} + \frac{1}{6}\left(\frac{1 + d_x}{2}\right)^2 \Delta x^2 \frac{\partial u^3}{\partial x^3} + ... \tag{91}$$

The second partial derivative term is significantly influenced by the difference in the grid spacing. For a uniform grid this term is zero and the truncation error of the finite difference approximation is second order. For nonuniform grid this term is non-zero and the derivative becomes first order accurate. Thus, any jump in grid spacing reduces the order of accuracy of the numerical system, and the larger the jump the larger the truncation error. In practice, the grid cannot be kept perfectly uniform for most cases, so minimizing the error by using gradual changes in grid spacing is preferred. The grid clustering also impacts the overset computations since the new denser grid regions have far more cells than the coarser background grid. This mismatch between grids leads to the computation of many more orphan nodes and thus reduces the quality of the information passed between grids. This problem may be avoided by either increasing the size of the boundary of the blade component grid, or by using a combination of rigid and elastic deformation to move the overset grid, as seen in Figure 16c. The latter is the most efficient method since additional grid points are not necessary. To extract the rigid component of the motion from the elastic data, a least-squares line-fit of the flapping and lead-lag deformations is applied, as illustrated in Figure 17. These lines become the new reference lines for the data effectively subtracting out the rigid component of the motion. The remaining elastic deflections that are applied now correspond to dx' instead of dx, which has a much smaller impact on the volume grid. The angle between the old and new reference lines is computed and this angle is used as a rigid deflection of the entire blade volume grid after the minimized elastic deformation has been applied. This data conversion represents a conversion from a simplified model to a more complex model. A reverse conversion is performed for the aerodynamic data from a more complex grid surface model to a simplified lifting-line model.

The conversion of the aerodynamic data from grid to lifting-line format involves computing the forces on each surface grid cell and decomposing these forces in the inertial

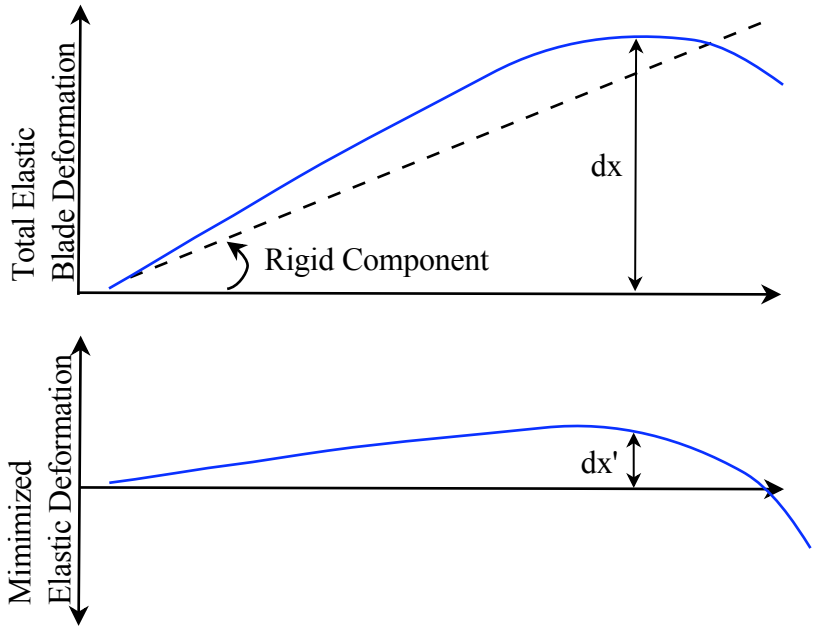


Figure 17: Illustration of method to extract rigid component of elastic data. Minimizes the amount of surface grid movement with respect to the surrounding volume grid

frame. Two methods have been evaluated for this fluid-structure interface. The first is a segment method where the blade grid is divided into segments such that the center of each segment corresponds to a point in the lifting-line model. These decomposed forces are then integrated over each grid segment to find the total forces and moments acting on the segment. The second implemented method is a slice method. This method takes a line segment through each surface cell that passes over the radial station at which the integration is taken. The reason for implementing both methods is to test whether there is a difference in including the local 3D effects on the blade in the integration instead of assuming that the forces at the radial station represent the forces along the entire segment.

The segment summation method is the more complex of the two methods. One difficulty inherent in this method is that unstructured cells will not necessarily lie on a single blade segment. Thus a simple summation of whole surface elements is not possible. This

issue can be addressed by computing the percent of the cell area present within each segment, then using only that fraction of the cell loads in the integration of each segment. The first computation is for the local forces on each cell face using

$$\vec{f} = c_p \vec{X}_{norm} A_{cell} p_{frac} \tag{92}$$

where A_{cell} is the area of the cell face on which the force is computed, p_{frac} is the fraction of that area that lies within the integration region and $\vec{X}_{norm}$ is the unit vector representing the orientation of the segment relative to the inertial frame. The final integration using these computations is

$$\begin{aligned} F_x &= \textstyle\sum_{i=1}^{N_{face}} f_x, \\ F_y &= \textstyle\sum_{i=1}^{N_{face}} f_y, \\ F_z &= \textstyle\sum_{i=1}^{N_{face}} f_z, \\ M_x &= \textstyle\sum_{i=1}^{N_{face}} \left(f_z \left(y - y_{mc} \right) - f_y \left(z - z_{mc} \right) \right), \\ M_y &= \textstyle\sum_{i=1}^{N_{face}} \left(f_x \left(z - z_{mc} \right) - f_z \left(x - x_{mc} \right) \right), \end{aligned} \tag{93}$$

and

$$M_z = \sum_{i=1}^{N_{face}} \left(f_y \left(x - x_{mc} \right) - f_x \left(y - y_{mc} \right) \right)$$

where N_{face} is the number of faces needed to cover the entire blade segment, $\vec{X}$ is the center point of the cell face, and $\vec{X}_{mc}$ is the quarter chord location related to the radial station of the cell face. If the cells are sufficiently small then the p_{frac} fraction can be neglected thus applying the entire load contribution of each cell whose center lies in the segment of interest to the integration. This method does not conserve the loads on each segment, but may be acceptable under certain circumstances since the error in the integration should balance out because there are overlapping cells on both sides of the segment division. However, the uneven nature of the unstructured grid surface, especially near the center of the chord length where the cells are larger, makes application of this assumption impractical without an overly fine blade surface that may exceed the requirements needed for a good CFD solution. This point has been illustrated in more detail in reference [67]. Dividing the cells

according to the percent area conserves the loads present within each segment and is a more general, and ultimately more robust, algorithm.

The slice method is a more commonly used method, especially in structured CFD codes where a cartesian mapping of the blade surface to I,J,K space makes this type of integration simple. This method takes line segments through the unstructured surface cells that overlap specified radial stations and integrates over these line segments. First the local forces in the inertial frame on each segment are computed using

$$\vec{f} = c_{ps}\vec{X}_{norm}l_{seg}. \tag{94}$$

Where c_{ps} is the average of the pressure coefficients at the adjoining nodes on the edges of the segment expressed by

$$c_{ps} = \left(\left(w_1 c_{pnode1} + w_2 c_{pnode2}\right)/2 + \left(w_3 c_{pnode2} + w_4 c_{pnode3}\right)/2\right)/2, \tag{95}$$

l_{seg} is the length of the segment on which the force is computed, w_i are the weight coefficients representing precisely how close the sliced segment is to each cell node, and $\vec{X}_{norm}$ is the unit vector representing the orientation of the segment relative to the inertial frame. Currently, w_i are all set to one because there is no advantage gained by taking time to compute them exactly. The final integration using these computations is

$$\begin{aligned} F_x &= \textstyle\sum_{i=1}^{N_{seg}} f_x, \\ F_y &= \textstyle\sum_{i=1}^{N_{seg}} f_y, \\ F_z &= \textstyle\sum_{i=1}^{N_{seg}} f_z, \\ M_x &= \textstyle\sum_{i=1}^{N_{seg}} \left(f_z\left(y - y_{mc}\right) - f_y\left(z - z_{mc}\right)\right), \\ M_y &= \textstyle\sum_{i=1}^{N_{seg}} \left(f_x\left(z - z_{mc}\right) - f_z\left(x - x_{mc}\right)\right), \end{aligned} \tag{96}$$

and

$$M_z = \sum_{i=1}^{N_{seg}} \left(f_y\left(x - x_{mc}\right) - f_x\left(y - y_{mc}\right)\right).$$

Where N_{seg} is the number of segments needed to slice the blade at a single radial station, $\vec{X}$ is the center point of the segment, and $\vec{X}_{mc}$ is the quarter chord location. This method

neglects any local 3D effects on the blade segments and is thus more of an approximation than the segment method. However, in the limit as the spacing between radial stations approaches zero the segment method is identical to the slice method. In application to this research, the maximum spacing between radial stations is 0.98 ft and the minimum spacing is 0.067 ft. When comparing the slice and the segment integration methods directly, the airloads predicted using each method are identical. Thus, in this case choosing one method over the other is a matter of computational efficiency rather than solution quality. The choice of method used for the results shown in the following chapters are mixed between the segment method using cell load conservation and the slice method. An input file flag enables toggling between the two methods during the coupling computations.

3.1.2.1 Delta Loads Coupling

The coupling method used between the CSD code and the CFD code is the delta loads method [66]. This method is a modified loose coupling scheme that includes aerodynamic damping in the structural computations, and is shown in Figures 18 and 19. The only way in which this type of coupling is different from the CSD-FW coupling is the way in which the airloads data are passed. The CSD code uses the converged airloads data computed by the CFD code over an entire revolution. This data remains constant since the data are computed before the CSD code begins its computations. The consequence is that the CSD code computations that use this aerodynamic data lack aerodynamic damping, which increases the number of revolutions needed to reach a converged solution as well as complicating the trim algorithm which requires that the airloads respond to the structural deformation in order to reach the target hub airloads. In order to facilitate the application of the trim algorithm, the delta loads, a modified version of the CFD loads, are passed to the CSD code each time. These loads are computed as

$$F^n_{tot} = F^n_{csd} + (F^{n-1}_{cfd} - F^{n-1}_{csd}). \tag{97}$$

Where F represents the six forces and moments acting on each blade segment. This form

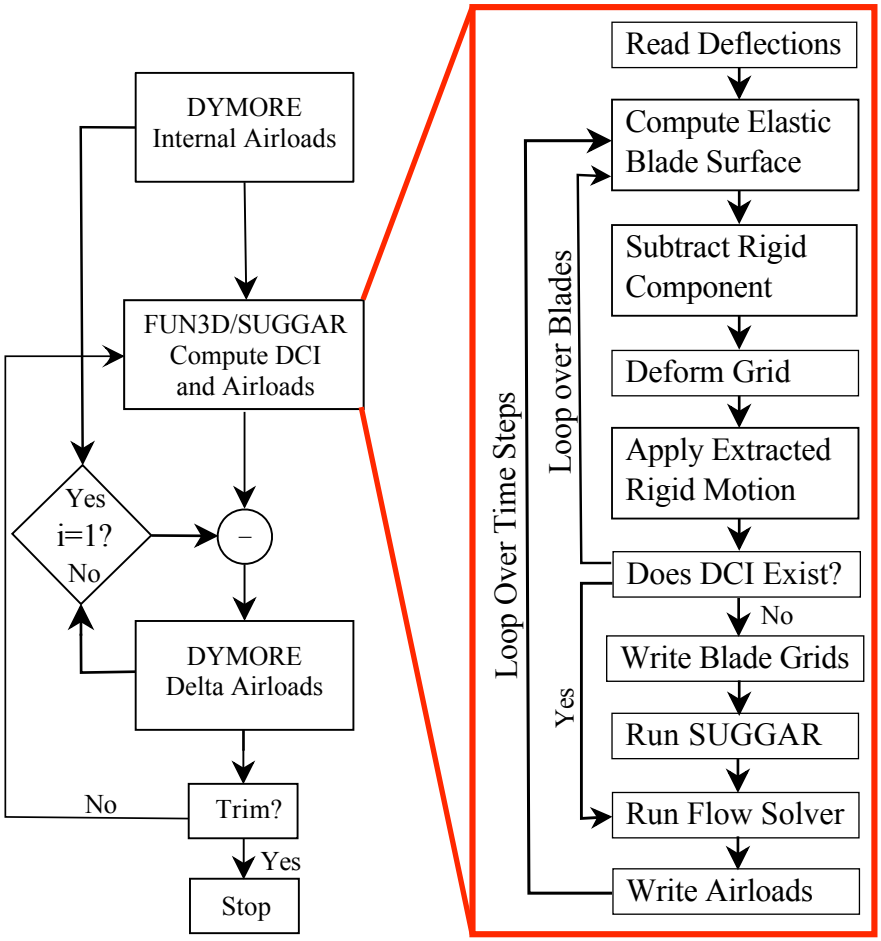


Figure 18: CFD-CSD coupling algorithm with concurrent DCI and airloads computations

of the airloads adds aerodynamic damping to the solution by using non-zero CSD loads at each step. When convergence has been reached, the CSD loads cancel out in equation 97, leaving only the CFD loads being applied to the structural model.

The delta loads coupling method is integrated with the component data conversions from lifting-line to grid format and vice versa to create the entire CFD-CSD coupling method. The first step in this method is to compute the initial blade deflections in DYMORE using an internal airloads model. These blade deflections are used to initialize the blade deflections in FUN3D for the first FSI step. In order to perform overset computations, FUN3D requires a set of DCI files classifying the nodes within the grids used for

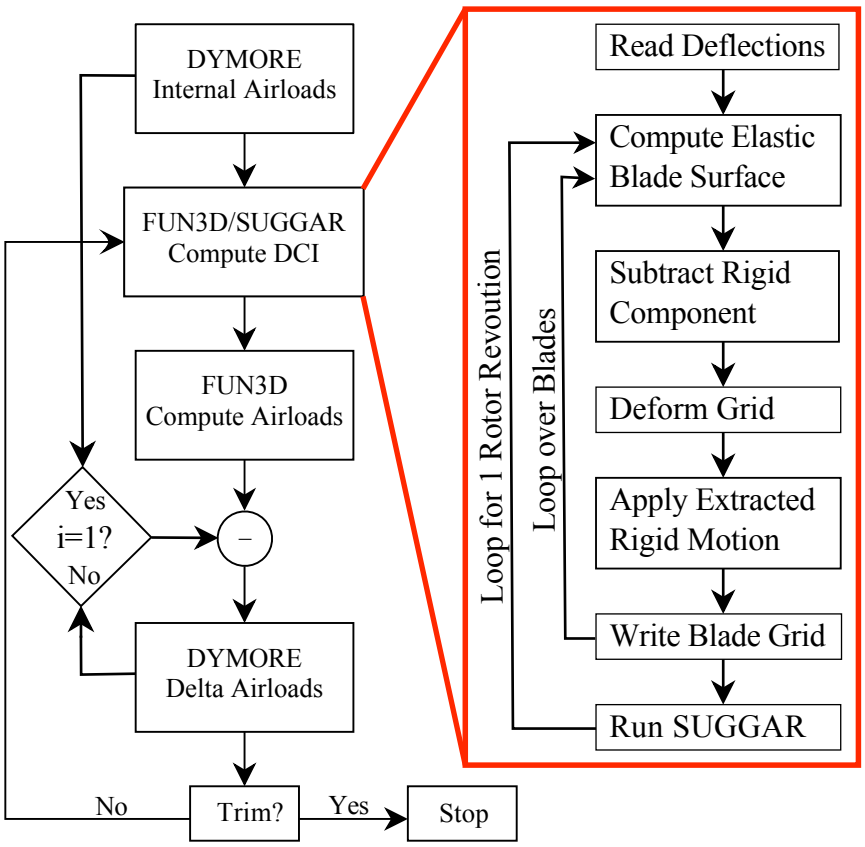


Figure 19: CFD-CSD coupling algorithm with independent DCI and airloads computations

interpolation. Since the exact state of the volume grid is not known ahead of time, this file must be written after the volume surrounding the blade surface has been deformed. Thus the hole cutting code SUGGAR, must be called internally from within FUN3D. This internal computation permits two methods to obtain the airloads from FUN3D. The first method is to compute the DCI files while the flow solver is running. This combines the DCI and airloads computations, as shown in Figure 18, into one CFD step. This method can be run for multiple revolutions since there is an internal checker to see whether the DCI file exists, if it exists, the SUGGAR step will be skipped, if it does not exist SUGGAR will be called to compute the DCI file. An alternate method is to compute all of the DCI files before the flow solver is run, as illustrated in Figure 19. This method requires no modifications to the

code, only the use of a command line option, which turns off the flow solver, but computes all of the grid motion for each step and stores the DCI Files. Since the DCI file existence checker is present when the flow solver is turned back on SUGGAR will never be called.

Both the concurrent and the separate DCI computation methods yield identical results, thus the choice of methods depends entirely on the availability of resources. Running the flow solver and grid deformation concurrently requires more time and more memory. If these are not available in sufficient quantity it is better to run these separately. The advantage to running these together is that if the case fails after a few time steps, DCI file generation will cease before the time to compute the other DCI files for the remainder of the revolution is wasted. Also, a concurrent run requires less interaction from the user and thus is more convenient to run than the separated computations.

3.2 Free Wake Model

3.2.1 Elastic Blade Motion

The original rigid, partially-articulated code has been enhanced to allow for elastic blade motion. The first change is the model itself. The original grid assumes a blade of the form $b\,(x, y_0)$, where b are the blade control point positions, y_0 is a constant corresponding to the blade quarter chord position, and x is the constant distribution of the blade control points along the span. This type of model does not allow for different blade geometries that have nonlinear quarter chord locations such as blades with tip sweep or tabs. The new model has been modified to take the form $b\,(x(t), y(t), z(t))$. This new model allows for nonlinear rigid blade geometries as well as the possibility of elastic blade motion during the rotor rotation. This change to the blade model requires the modification of many of the computations related to blade geometry in the code. Originally, the influence coefficient matrices shown in equation 54 were computed at time zero then used as constant matrices for all following time steps. For a rigidly deforming blade this is both a correct and an efficient method for computing the induced velocity relating the bound and near-wake vortices to the blade

control point locations. However, when the blade deforms, this relation is no longer a constant. Thus, new relations must be computed each time the blade deforms. This is not as efficient, since additional computations of the Biot-Savart law over all of the near wake and bound vortices are required at each time step, but it is necessary to allow for elastic blade motion.

The governing equations for the free wake geometry are in the form of the predictor-corrector (PC2B) scheme discussed previously. These equations of motion require no modification since they do not depend on the blade geometry. However, the boundary condition that determines the position of the attachment of the wake trailers to the blade must be modified to allow for elastic motion. The blade attached boundary condition at $\xi = 0$ is a function of the blade geometry at azimuthal position ψ. The new expression for this boundary condition is $r_{\xi=0,\psi} = b\,(x(t), y(t), z(t))$, where the blade motion now varies spatially and over time (azimuth). This concept is illustrated in Figure 20, where the top image plots the wake trailer locations over the entire rotor revolution using the elastic motion, and the lower image provides the rigid wake trailer locations for comparison. The computations of flap and pitch motion that were implemented in the original code are not necessary for CSD coupling. The fully-articulated motion is computed by the CSD code, thus these angles are obtained from the data computed by the CSD code. This structural data includes an additional lead-lag motion which was not included in the free wake computations before. Since the pitch angles are input directly using the rotations computed by the CSD model, the trim computations from the CSD code are now implicitly included in the free wake code. All of these changes reduce the number of assumptions made by the original code and allow the free wake model to follow the motions of the structural model with no restrictions. This unrestricted blade motion allows the free wake geometry to conform to a deformable blade geometry, and thus predicts a wake geometry based on a more realistic system.

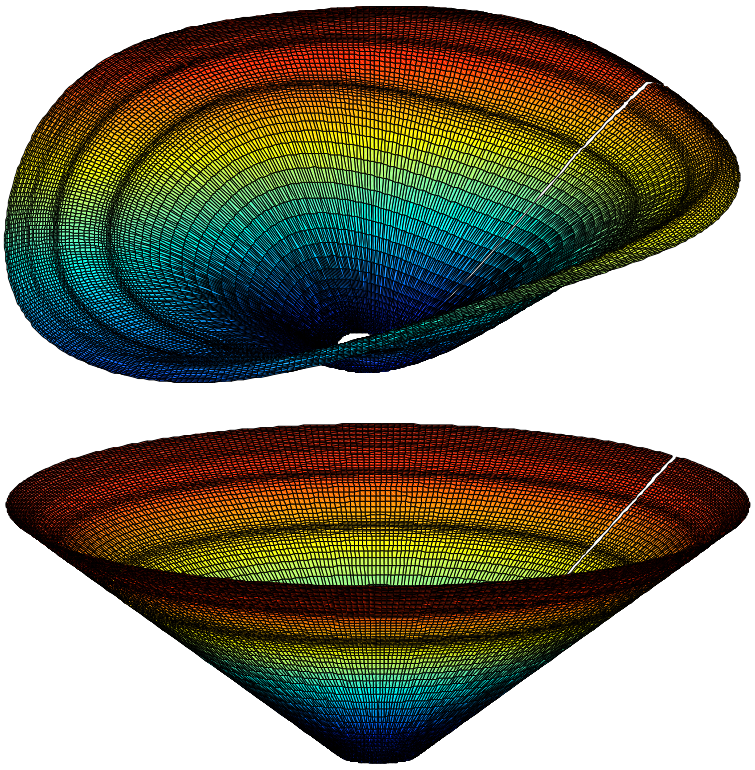


Figure 20: Illustration of impact to blade fixed wake trailer location using rigid vs. elastic geometry

3.2.2 Computational Timing Improvements

The computation time-intensive nature of free wake codes makes any time saving device worthwhile. Part of the problem lies in the way in which the induced velocities are computed in vortex filament models. In these types of codes, the wake is represented by sets of constant strength vortex filaments connected in series. Each vortex filament represents a discretized segment of the physical vortex wake, which in reality has a strength and a curved geometry that are continuously changing. The velocity field of each filament influences every point in the flow field. The summation of all the induced velocity contributions of all the vortices produces the total induced velocity field surrounding the rotor. However, the entire field is not of interest, only discrete points located on the rotor plane corresponding to the blade control points at each time step are of interest. To find the induced velocity

of these points, the velocity contribution of each filament to each control point must be computed using the Biot-Savart law defined in equation 53. Since each vortex filament is placed differently relative to the control points, each vortex has a different contribution to the induced velocity at each point. Thus this contribution must be computed separately for each vortex filament and must be repeated for each control point for which the induced velocity is sought. The resulting expression for the velocity at a specific control point in the field becomes

$$\vec{V} = \sum_{i=1}^{N_{fil}} \vec{V}_{biot}\vec{r}. \tag{98}$$

Where $\vec{V}_{biot}$ is the velocity computed using equation 53, N_{fil} is the number of filaments in the system, and $\vec{r}$ is the vector distance of each filament to the point of interest. This is an inherently time-consuming process, which may be improved upon by applying various computational methods.

Computational methods for improving efficiency include changing the way in which the cases are run, or by changing the codes themselves. One example of changing the cases involves data passing intervals. Tight coupling is the most general method for passing data, which can work for steady-state and maneuvering cases alike. However, running the free wake code at a small enough time step to accommodate the structural code can add significant computation time. Instead, loose coupling, which is specific to steady-state cases only, can be used to coarsen the free wake model, while leaving the structural model unchanged. This loose coupling method shows enormous timesaving of about 85% for a one wake trailer run as opposed to the time needed for a one wake trailer tightly coupled run. Modifications to the source code can also be used to save time. Some of these modifications include removing unused code, removing most of the write statements to unused output files, replacing as many instances of trigonometric functions as possible, which are time-consuming relative to arithmetic operations, and swapping the array storage indices so that nested loop searches are faster. An additional possibility that has not been demonstrated here is the application of the fast multipole method. This method has been shown

to accelerate the computations needed for wake trailer computations by using a multipole approximation to define the far field [68]. This method has been demonstrated within the free wake code CHARM [69]. These methods are only a subset of the possibilities. Further code modifications can be made that take advantage of parallel processing. This type of improvement will eliminate most of the run time dependency on the number of wake trailers.

3.2.2.1 Parallel Free Wake Computations

The free wake code used in this project is inherently parallelizable, since the majority of the run time is spent computing the induced velocities using the Biot-Savart law. As described earlier, this relation computes the contribution of each vortex segment to the inflow at one point in space. This computation is then repeated for the same point in space for every vortex segment in the model to obtain the total flow velocity at that point. For multiple points in space the whole process is repeated for each point. The resulting code structure is illustrated in Figures 21 and 22 for both the serial and parallel algorithms respectively. Each added wake trailer increases the number of vortex segments in the model, which increases the number of computations exponentially, thus increasing the run time of the serial code exponentially. The parallel version of the code will divide the computations for each wake trailer over multiple processors. This is accomplished by replacing the wake trailer loop control with a processor dependent version, then passing the computed data amongst all processors. The wake trailer looping index is modified for parallel computations using the processor rank in the expression

$$w = (rank + 1)x - (x - 1), (rank + 1)x, \tag{99}$$

where

$$x = \frac{Number\ of\ Wake\ Trailers}{Number\ of\ Processors}. \tag{100}$$

This is a general formulation where any number of wake trailers can be divided among any number of processors, as long as the result is an integer. This is useful if the user wants to

run a dense grid in parallel when not many processors are available.

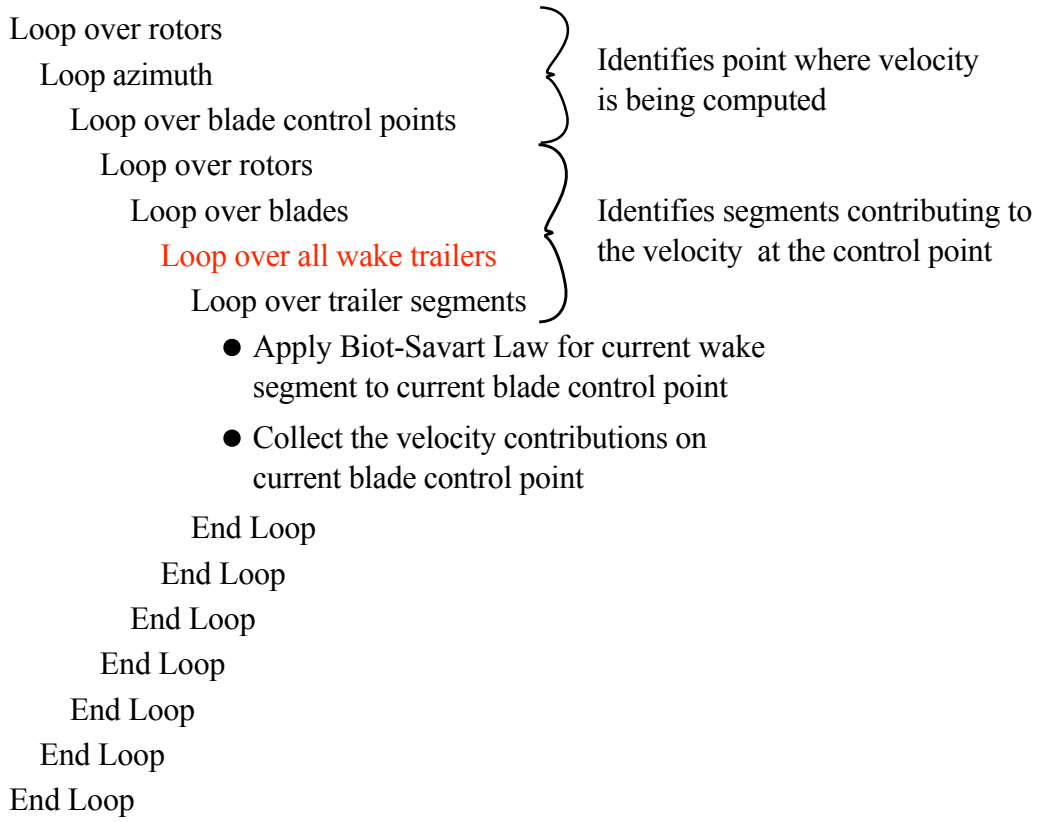


Figure 21: Nested loop structure of the serial code

The end result of running this divided loop is that each processor will hold the inflow contributions on all blade segments for only the wake trailers it is assigned. In order to use this information, the data must be collected and distributed to all processors. This step is performed using MPI calls to wait until all processors reach the same point in the code and to then collect the data. Forcing the processors to all reach the same point ensures that each processor has completed its partial inflow summations before the data are used. Once the data are collected, the partial inflow values from all processors are summed together, then the resulting total inflow is distributed to all processors. Once the data have been distributed, the execution proceeds as before with each processor performing the same computations as the other processors. The method selected to do the parallelization was

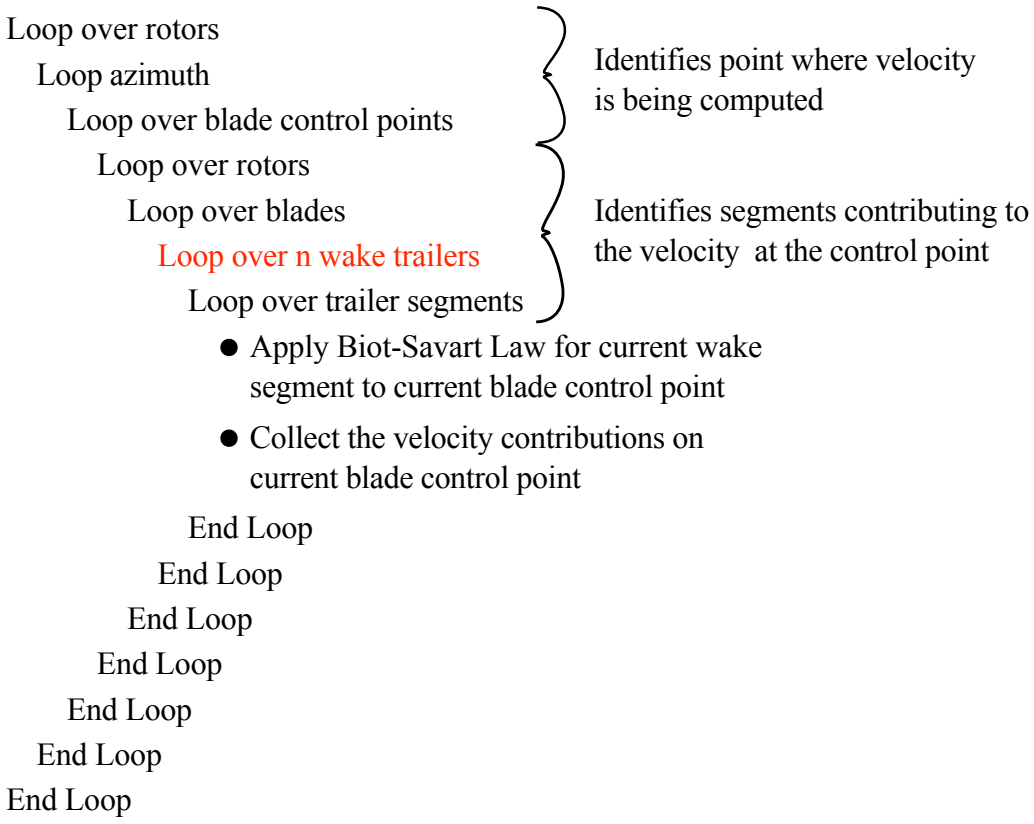


Figure 22: Nested loop structure of the parallel code

chosen to minimize the amount of communication between processors. The loop indices are a function of rank and thus involve no communication, the only passed information is through the data summation from all processors.

3.3 CFD Model

3.3.1 Rigid Rotor Motion

A rigid rotor motion algorithm to implement fully-articulated motion has been added to FUN3D. Rigid motion can be included in FUN3D by either specifying the Fourier coefficients, or by using an elastic data file containing rigid surface deflections. For rigid motion the Fourier coefficients are more efficient since the CFD code will not have to assume that the volume must be deformed elastically, and thus will not compute a new volume mesh

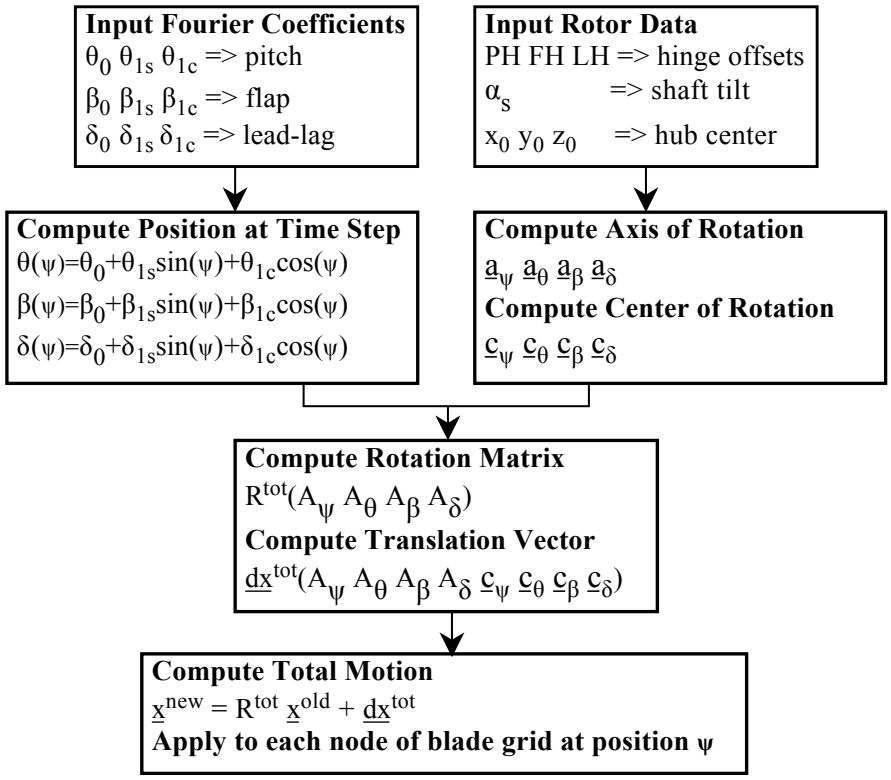


Figure 23: Flow chart describing the application of fully-articulated rigid motion in FUN3D

to surround the blade. The coefficient method moves all nodes in the entire overset blade grid by applying the rotation matrix and translation vector associated with the rigid motion. The algorithm for this method using the Fourier coefficient inputs is shown in Figure 23. This algorithm models flap, lead-lag, and pitch motions with the option of including hinge offsets and shaft tilt into the analysis. The method using elastic inputs has been developed separately and is described as the elastic coupling method in section 3.1.2.

The general procedure to compute the blade motion is to first compute the hinge angles at the current time step using Fourier coefficient reconstruction. The hinge offsets are included through the computation of the center of rotation of each degree of freedom. Finally, using the combination of center of rotation, axis of rotation, and angle of rotation, a set of transformation matrices are obtained. Through matrix multiplication these transformation matrices are combined to form one transformation matrix to specify the total

motion of each rotor blade. The motion data are then applied to the unstructured grid at each time step for each blade, resulting in fully-articulated rotor motion. The algorithm has been written to allow the possibility of multiple rotors, with any number of blades per rotor. A more detailed explanation is as follows, the basic theory of which can be found in reference [70]; the RBM equations are computed in steps. The first step is to determine the axes of rotation and their corresponding centers of rotation. These will be denoted as $\vec{a}_\eta$ and $\vec{c}_\eta$, where η is a placeholder for the rotation type of interest. The matrix that defines the rotation of the system about a specific axis will be denoted as A_η and is defined as

$$A_\eta = \begin{bmatrix} (1-\cos(\eta))a_{\eta x}^2 + \cos(\eta) & \dots \\ (1-\cos(\eta))a_{\eta x}a_{\eta y} + \sin(\eta)a_{\eta z} & \dots \\ (1-\cos(\eta))a_{\eta x}a_{\eta z} - \sin(\eta)a_{\eta y} & \end{bmatrix}$$

$$\begin{matrix} (1-\cos(\eta))a_{\eta x}a_{\eta y} - \sin(\eta)a_{\eta y} \\ (1-\cos(\eta))a_{\eta y}^2 + \cos(\eta) \quad \dots \\ (1-\cos(\eta))a_{\eta x}a_{\eta y} + \sin(\eta)a_{\eta x} \end{matrix} \tag{101}$$

$$\begin{bmatrix} (1-\cos(\eta))a_{\eta x}a_{\eta z} + \sin(\eta)a_{\eta y} \\ (1-\cos(\eta))a_{\eta y}a_{\eta z} - \sin(\eta)a_{\eta x} \\ (1-\cos(\eta))a_{\eta z}^2 + \cos(\eta) \end{bmatrix}.$$

The centers of rotation and axes of rotation include the contributions of hinge offsets and shaft tilt as follows,

$$\begin{aligned} \vec{a}_\psi &= (-\sin(-\alpha_s)sn, 0.0, \cos(-\alpha_s)sn), \\ \vec{c}_\psi &= (x_0, y_0, z_0), \\ \vec{a}_\theta &= (\cos(-\alpha_s)sn, 0.0, \sin(-\alpha_s)sn), \\ \vec{c}_\theta &= (x_0 + PH\cos(-\alpha_s), y_0, z_0 + PH\sin(-\alpha_s)), \\ \vec{a}_\delta &= (-\sin(-\alpha_s), 0.0, \cos(-\alpha_s)), \\ \vec{c}_\delta &= (x_0 + LH\cos(-\alpha_s), y_0, z_0 + LH\sin(-\alpha_s)), \\ \vec{a}_\beta &= (0.0, -1.0, 0.0), \end{aligned} \tag{102}$$

and

$$\vec{c}_\beta = (x_0 + FH\cos(-\alpha_s), y_0, z_0 + FH\sin(-\alpha_s))$$

where $\vec{X}_0$ is the hub center, PH, FH, LH are the pitch, flap, and lag hinges of the rotor, and α_s is the shaft tilt angle. The variable sn is either 1 or -1 depending on the direction of rotation of the rotor. The angles are computed by reconstructing user input Fourier coefficient data as

$$\begin{aligned} \theta &= \theta_0 + \theta_{1s}\sin(\psi) + \theta_{1c}\cos(\psi), \\ \beta &= \beta_0 + \beta_{1s}\sin(\psi) + \beta_{1c}\cos(\psi) + \quad \ldots, \end{aligned} \tag{103}$$

and

$$\delta = \delta_0 + \delta_{1s}\sin(\psi) + \delta_{1c}\cos(\psi) + \quad \ldots.$$

Where the flap and lead-lag are defined using up to four harmonics and the pitch is only defined using one harmonic. The total rotation matrix is then constructed as follows,

$$\begin{aligned} R' &= A_\theta I, \\ R'' &= A_\delta R', \\ R''' &= A_\beta R'', \end{aligned} \tag{104}$$

and

$$R^{tot} = A_\psi R'''$$

where I is the identity matrix, R', R'', R''' are intermediate steps, and R^{tot} is the total rotation matrix. The corresponding displacements are computed as

$$\begin{aligned} \vec{dx} &= -A_\theta(\vec{c}_\theta) + \vec{c}_\theta, \\ \vec{dx}'' &= A_\delta(\vec{dx}' - \vec{c}_\delta) + \vec{c}_\delta, \\ \vec{dx}''' &= A_\beta(\vec{dx}'' - \vec{c}_\beta) + \vec{c}_\beta, \end{aligned} \tag{105}$$

and

$$\vec{dx}^{tot} = A_\psi(\vec{dx}''' - \vec{c}_\psi) + \vec{c}_\psi$$

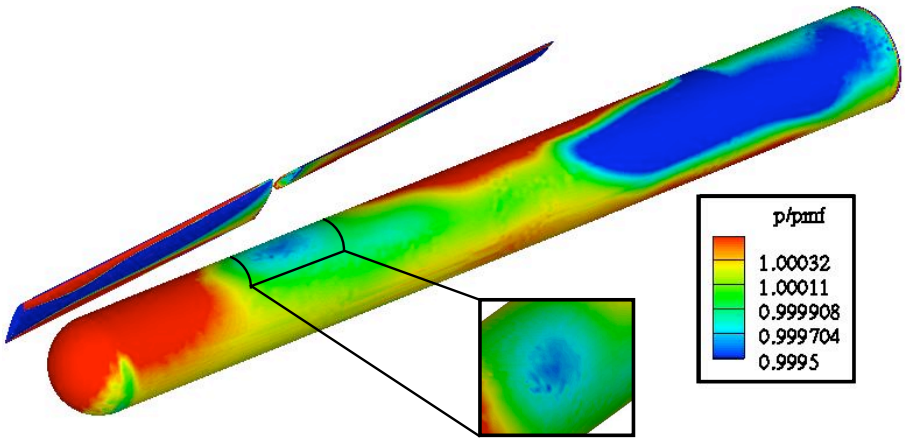


Figure 24: Image illustrating the presence of numerical stiffness in the compressible equations

where $\vec{dx}', \vec{dx}''$, and $\vec{dx}'''$ are intermediate steps, and $\vec{dx}_{tot}$ is the resulting displacement vector. These components are used to relate the initial point to the deformed point as follows,

$$\vec{x}^{new} = R^{tot}\vec{x}^{old} + \vec{dx}^{tot}. \tag{106}$$

This expression is applied to every grid point in the blade component grid being moved. The solution of the URANS equations then proceeds as before using the deformed grid and the grid velocities computed from the previous to the current grid configurations.

3.3.2 Low Mach Number Preconditioning

Using the compressible RANS equation in low Mach number regimes results in high equation stiffness, which adds nonphysical fluctuations to the flow solution, as illustrated in Figure 24. This high stiffness is the result of the large variation in the Mach number regimes seen in the simulation, which result in a numerical system with large differences in the eigenvalues, and thus large condition numbers. However, it is not always appropriate to use the incompressible equations if compressible regimes are included in the simulation. These mixed Mach regimes require the advantages of both systems of equations. The

purpose of unsteady low Mach number preconditioning (ULMP) is to be able to use the compressible equations when the Mach number exceeds 0.3 in some locations, but to improve the solution when there are also regions that have Mach numbers below 0.3. The magnified region shows the fluctuations in the vortex impact on the surface of the fuselage. In reality, this impact should have a smoother profile as will be illustrated further in chapter 6. Low Mach number preconditioning is designed to reduce this equation stiffness and thus improve the quality of the solution. Originally, FUN3D was developed to model fixed-wing, steady-state cases. This development includes many of the analyses such as low Mach number preconditioning. However, many of these methods have not been extended to time-accurate analysis. For rotor cases, the Mach number can be smaller than for fixed-wing cases, because the velocity over the blades is increased through blade rotation and not through increased forward flight speed. Thus the flow speed over the fuselage, and over portions of the rotor blades can be close to zero, and thus the need for low Mach number preconditioning is more important. This option has been extended to include the possibility of computing time-accurate cases, which is necessary for moving rotor computations. Low Mach number preconditioning is implemented using the method detailed in reference [71]. In the steady-state case, only the spatial terms have been preconditioned since the time terms are not present. However, for the time-accurate case, reintroducing the preconditioned time term into the governing equations will not produce a valid set of governing equations, since the addition of the preconditioning matrix to the time term destroys time-accuracy. Instead, the time-accurate method is implemented using dual time-stepping [72]. This method preserves time-accuracy by adding the preconditioning matrix (Γ) to the pseudo-time term of the governing equations instead of to the time term. This allows the coefficient of the time term to remain unchanged and thus will not change the time step of the system. The modified time-accurate governing equations are given as

$$\Gamma V \frac{Q^{m+1} - Q^m}{\Delta\tau} + \frac{V}{\Delta t}(\phi_{m+1}Q^{m+1} + \phi_n Q^n + \phi_{n-1}Q^{n-1} + \phi_{n-2}Q^{n-2} + \dots) = R^{m+1} \qquad (107)$$

where the preconditioning matrix in terms of conservation variables takes the form

$$\Gamma = \left(M + \Theta \vec{u}\vec{v}^T\right) M^{-1}, \tag{108}$$

where

$$\Theta = \frac{1}{c^2}\left(\frac{1}{\beta} - 1\right), \tag{109}$$

$$\vec{u} = \begin{bmatrix} 1 & u & v & w & H \end{bmatrix}^T, \tag{110}$$

$$\vec{v} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T, \tag{111}$$

H is the total enthalpy per unit mass of the form

$$H = \frac{c^2}{\gamma - 1} + \frac{\left(u^2 + v^2 + w^2\right)}{2}, \tag{112}$$

and β is a preconditioning parameter computed as

$$\beta = \min\left(1, \max\left(M^2, KM_\infty^2\right)\right) \tag{113}$$

where K is a constant, usually set to three in the literature, but here the quantity KM_∞^2 is set by the user in the input file. This is the original input setting, and it allows the user to change the quantity KM_∞^2 to any expression, and is thus more general. KM_∞^2 is an empirical quantity which has been found to produce suitable steady-state preconditioned results, but for unsteady preconditioning other quantities are used, and will be discussed later. The preconditioning matrix must be transformed into conservative variables in order to match the form of the solution used in the governing equations. This transformation is implemented using the transformation matrix M, which contains the relation between the conservative variables and the primitive variables, and is of the form

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ u & \rho & 0 & 0 & 0 \\ v & 0 & \rho & 0 & 0 \\ w & 0 & 0 & \rho & 0 \\ \frac{\left(u^2+v^2+w^2\right)}{2} & u\rho & v\rho & w\rho & \frac{1}{\gamma-1} \end{bmatrix}, \tag{114}$$

and its inverse M^{-1} is

$$M^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -\frac{u}{\rho} & \frac{1}{\rho} & 0 & 0 & 0 \\ -\frac{v}{\rho} & 0 & \frac{1}{\rho} & 0 & 0 \\ -\frac{w}{\rho} & 0 & 0 & \frac{1}{\rho} & 0 \\ \frac{(\gamma-1)}{2}\left(u^2+v^2+w^2\right) & -u(\gamma-1) & -v(\gamma-1) & -w(\gamma-1) & (\gamma-1) \end{bmatrix}. \tag{115}$$

The inviscid Roe flux for this new system of governing equations is derived as

$$|A_V|(V_R - V_L) = \frac{|\Delta\xi|}{J}\begin{bmatrix} \alpha_4 \\ \tilde{u}\alpha_4 + k_x\alpha_5 + \alpha_6 \\ \tilde{v}\alpha_4 + k_y\alpha_5 + \alpha_7 \\ \tilde{w}\alpha_4 + k_z\alpha_5 + \alpha_8 \\ \tilde{H}\alpha_4 + \tilde{\bar{u}}\alpha_5 + \tilde{u}\alpha_6 + \tilde{v}\alpha_7 + \tilde{w}\alpha_8 - \frac{\tilde{c}^2}{\gamma-1}\alpha_1 \end{bmatrix} \tag{116}$$

where

$$\alpha_1 = |\tilde{\bar{u}}|\left(\Delta\rho - \frac{\Delta p}{\tilde{c}^2}\right), \tag{117}$$

$$\alpha_2 = \frac{1}{2\tilde{c}^2}|\tilde{\bar{u}}' + \tilde{c}'|\left(f^+\Delta p + \tilde{\rho}\tilde{c}\Delta\tilde{u}\right), \tag{118}$$

$$\alpha_3 = \frac{1}{2\tilde{c}^2}|\tilde{\bar{u}}' - \tilde{c}'|\left(f^-\Delta p - \tilde{\rho}\tilde{c}\Delta\tilde{u}\right), \tag{119}$$

$$\alpha_4 = \alpha_1 + \frac{2}{\tilde{\beta}\left(f^+ + f^-\right)}\left(\alpha_2 + \alpha_3\right), \tag{120}$$

$$\alpha_5 = \frac{2}{\left(f^+ + f^-\right)}\tilde{c}\left(f^-\alpha_2 - f^+\alpha_3\right), \tag{121}$$

$$\alpha_6 = |\tilde{\bar{u}}|\tilde{\rho}\left(\Delta u - k_x\Delta\bar{u}\right), \tag{122}$$

$$\alpha_7 = |\tilde{\bar{u}}|\tilde{\rho}\left(\Delta v - k_y\Delta\bar{u}\right), \tag{123}$$

and

$$\alpha_8 = |\tilde{\bar{u}}|\tilde{\rho}\left(\Delta w - k_z\Delta\bar{u}\right) \tag{124}$$

where the $\tilde{(.)}$ notation indicates the Roe-averaged state of the variable, $\bar{u}$ is the contravariant velocity normal to the cell interface, and $k_{x,y,z}$ are the direction cosines.

There are other low Mach number preconditioning methods that specifically reference extensions to unsteady cases [73]. This method proceeds as the one that was just described. However, in this method the preconditioning parameter β is replaced with

$$\beta = \min\left(1, \max\left(M^2, M_s^2, M_u^2\right)\right). \tag{125}$$

Where M_s is the steady Mach number limit defined as KM_∞^2, and M_u is the unsteady Mach number limit

$$M_u = \frac{M_\infty L}{\pi C \Delta x}, \tag{126}$$

where L is an unsteady reference length scale, C is the CFL number based on propagation velocity, and Δx is a characteristic grid dimension. The main difference between this unsteady modification and the previous method is that the preconditioning parameter β is selected to be the larger of the steady and unsteady Mach numbers. This modification is applied to take advantage of the convergence benefits of the unsteady Mach number and the accuracy characteristics of the steady Mach number, by choosing the one that uses the local Mach number to compute β over a larger portion of the grid. Further modifications described in reference [73] include changes to the artificial dissipation terms of the spatial discretization, by replacing the eigenvalue diagonal matrix with a matrix which applies the steady spectral radius to the continuity equation and the unsteady spectral radii to the momentum and energy equations. This eliminates the problem of excessive damping in the flow field associated with using either unsteady or steady preconditioning terms alone. This modification has not been implemented in this research because similar dissipation terms do not exist within FUN3D. Since the steady preconditioning over-damps the pressure solution and the unsteady preconditioning over-damps the velocity and the temperature fields a combination of the two would yield the best solution. In general, the application of low Mach number preconditioning to time-accurate cases is dominated by the need to use the dual time-stepping formulation to preserve time-accuracy. Once this method has been applied, modifications to balance the benefits of the steady and unsteady preconditioning

parameters enhance the solution further.

Since the parameter β is a function of the local Mach number, compressible phenomena such as shock waves that occur at high Mach numbers may still be modeled. At high Mach numbers β is set to one, this turns off the preconditioning at the selected point, thus leaving high Mach regions unchanged. However, even though the effect of preconditioning does not explicitly modify high Mach regions, the large range of Mach numbers that would have to be present to cover the range of incompressible to transonic flow creates an unbalanced preconditioning matrix, which will introduce more numerical error into the solution than a completely low Mach number case would. The decision to use the LMP algorithm must be weighed against these considerations to see if any benefit would be achieved.

CHAPTER 4

TEST CASES

4.1 UH60A Rotor

The UH60A helicopter flight test database [74] has become the correlation standard for CFD-CSD coupling. This flight test was sponsored by NASA and the US Army and occurred in February of 1994. The instrumented aircraft has 242 pressure transducers along the blade as various radial stations. A wide variety of flight conditions were chosen throughout the flight envelope of the aircraft, and included both steady-state and maneuvering conditions. The subset of cases studied here include counter 8424, a moderate forward flight case, counter 8534, a high speed forward flight case with transonic effects on the advancing blade, and counter 9017 a high thrust forward flight case, which exhibits dynamic stall characteristics [74, 75, 76]. Both 9017 and 8534 are close to the boundaries of the flight envelope and thus are able to effectively test the limits of the solution methods presented. Case 8424 is closer to the center of the flight envelope and is thus more suitable for proof of concept studies. The specific parameters of these cases are shown in Table 2, and the associated trim targets are shown in Table 3.

Table 2: UH60A test case configurations

Condition Counter	Mod Speed 8424	High Speed 8534	High Thrust 9017
μ	0.304	0.368	0.237
C_T/σ	0.088	0.084	0.129
M_∞	0.194	0.236	0.157
M_t	0.642	0.642	0.665
α_s [deg]	-4.16	-7.31	-0.15
h [ft]	5757	3273	17133

Table 3: UH60A trim parameters

Condition Counter	Mod Speed 8424	High Speed 8534	High Thrust 9017
Thrust [lbs]	15962	17944	16688
Roll Mom. [ft lbs]	-4017	-6884	-320
Pitch Mom. [ft lbs]	-3442	-2583	112

4.1.1 CSD Model

The DYMORE model for the UH60A helicopter rotor is modeled using basic structural elements. The construction of this model can be seen in Figure 25. The main blade and root connector elements are modeled using Euler-Bernoulli beam elements. The blade has ten 3^{rd} order finite elements, the root connectors each have two 3^{rd} order finite elements, and the hinges are modeled using revolute joints. In DYMORE, the four-bladed model consists of not only the blades, but also flap, lead-lag, and pitch hinges, a hydraulic lead-lag damper [77], an equivalent spring, and a hub component to tie it all together. The equivalent spring is included to represent the stiffness of the pitch link and swash plate components, which have not been included. These have been left out because there is not sufficient data to model them correctly. The associated lifting lines used for the aerodynamic computations contain 64 evenly spaced points that lie along the blade quarter chord. The geometry of the blade itself includes both the swept tip and the tab, which are applied through the lifting line data. The structural elements have structural twist data but no offsets from the beam axis.

Since this model uses a nonphysical equivalent spring to represent the stiffness of the control linkages, the stiffness of this spring must be determined such that it represents as closely as possible the stiffness of the control system. However, there is no experimental measure of the stiffness of the physical components of this system. Therefore, the value is determined based on an evaluation of the structural model as compared to experimental data. The affect of the stiffness of this equivalent spring element on the system is compared

in Figures 26, 27, and 28 to the experimental data. In the following data the stiff spring stiffness is taken to be 1090 $ft - lbs/deg$ and the soft spring is 363 $ft - lbs/deg$. When comparing the results, the soft spring does have slightly better solution as compared to experiment on the retreating side of the rotor. Using the stiff spring does not improve the results beyond this point. In a more general sense, the significance of the measured airloads results show that the structural model is a good approximation to the physical system, since it is able to capture the main characteristics seen in the experimentally measured bending moments. However, the magnitude of these characteristics is somewhat low, owing to the approximate nature of the model, because not all aspects of the physical blades are known.

4.1.2 Free Wake Model

The MFW model for the UH60A rotor is constructed entirely of vortex filaments and control points. The vortex filaments are lined up along the blade quarter chord and along the wake trailers, as is seen in Figure 9. The blade control points are lined up at the blade vortex filament midpoints at the three quarter chord location. The blade twist and the chord offset information are stored at each of the control points, as are the intersection points of the blade attached vortex filaments. The distribution and number of wake trailers is variable, as is the distribution and number of radial stations. However, the wake trailer locations, although variable, must lie on one of the endpoints of the blade attached vortex filaments.

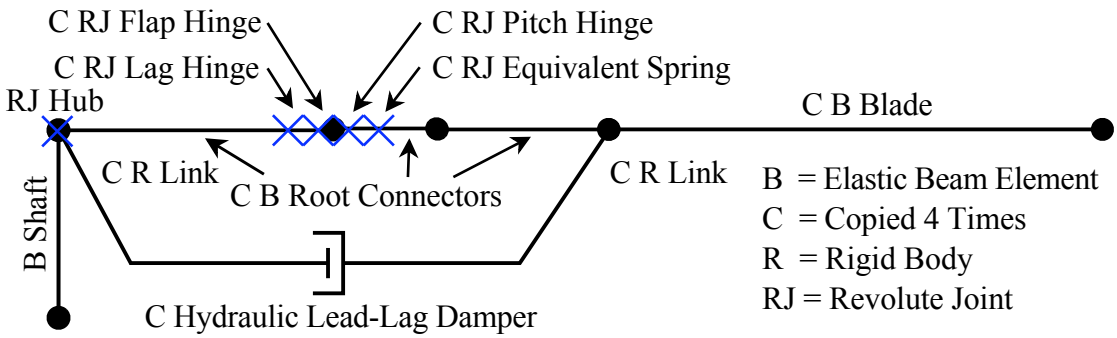


Figure 25: Illustration of DYMORE equivalent spring model (not to scale)

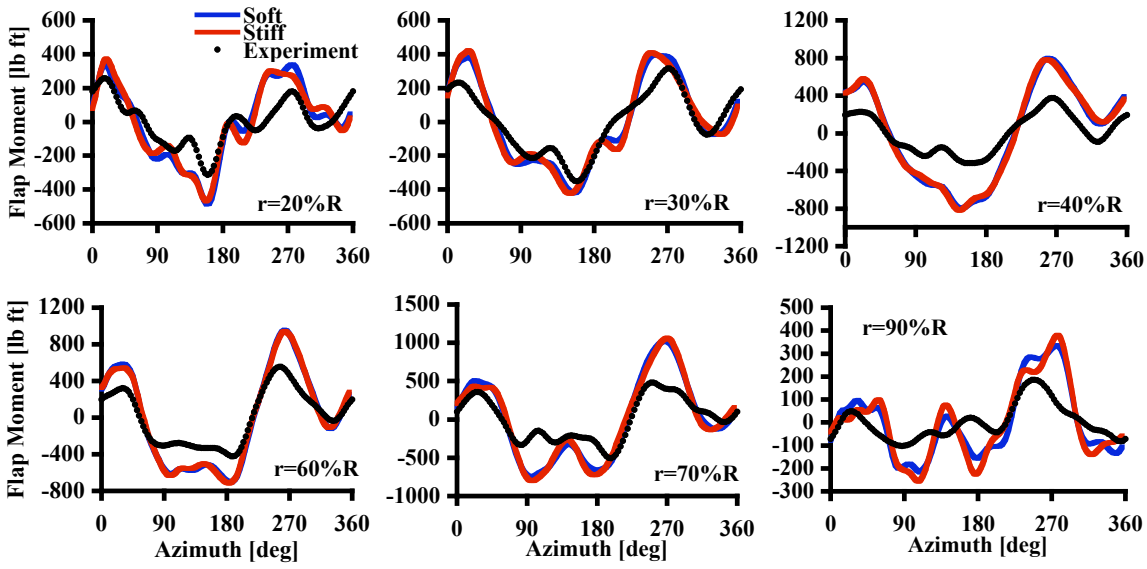


Figure 26: UH60A case 8534 comparison of the flap bending moment [ft lbs] using different equivalent spring stiffnesses computed using measured airloads

Specifically, the geometry of the UH60A blade includes the swept tip and the tab, and the 64 evenly spaced control point locations are chosen so that the blade attached vortex filaments match the lifting line used in the DYMORE model.

4.1.3 Unstructured CFD Model

The FUN3D model for the UH60A rotor is an unstructured overset grid constructed with tetrahedral elements. This grid, shown in Figure 29, consists of 1,726,159 nodes and 10,060,334 elements per blade grid. The background grid is a box with sides measuring 310 ft in length. This grid has 863,586 nodes and 5,069,055 elements, resulting in 7,768,222 nodes and 45,310,391 elements in the entire overset grid. The surface grid, shown in Figure 30, is constructed using triangular surface elements. The solver was run using 2^{nd} order spatial accuracy and hybrid $2^{nd}/3^{rd}$ order time accuracy. The Spalart-Allmaras turbulence model is used with a Venkatakrishnan limiter and a 1^{o} azimuthal step size with 20 subiterations. In addition to these settings, for cases where blade deformation is needed the elasticity solver is used to deform the volume surrounding the blade with a convergence

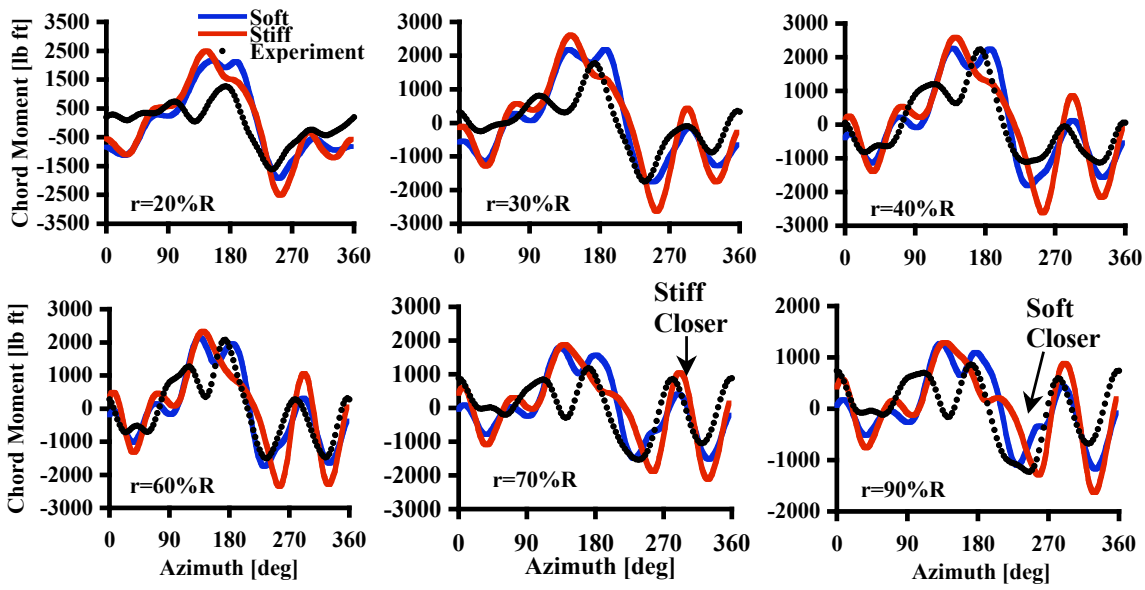


Figure 27: UH60A case 8534 comparison of the chord bending moment [ft lbs] using different equivalent spring stiffnesses computed using measured airloads

tolerance of 1e-5. A comparison of the airload results for a 1^o 20 subiteration case with a 0.5^o 20 subiteration case in Figure 31 show that using a 1^o time step size does not significantly change the results except around the sweep transition location where there is an improvement in the magnitude. Similar changes will be shown for adjustments to the structured CFD code in the next section.

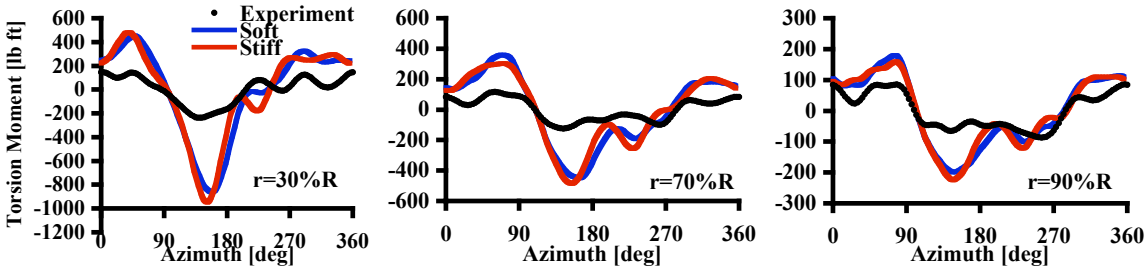


Figure 28: UH60A case 8534 comparison of the torsional moments [ft lbs] using different equivalent spring stiffnesses computed using measured airloads

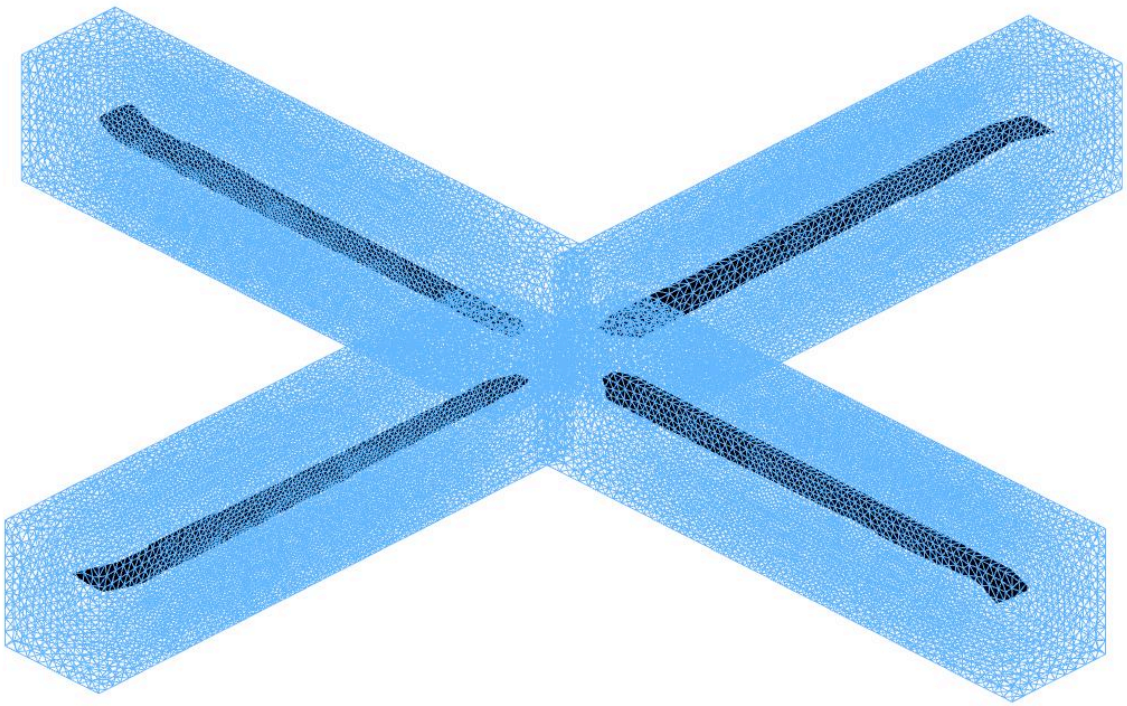


Figure 29: UH60A unstructured overset grid used in FUN3D

4.1.4 Structured CFD Model

The structured data provided by reference [3] uses OVERFLOW as the CFD solver. The UH60A geometry used to create the structured grid is identical to the geometry used to create the unstructured grid. The surface mesh for this grid is shown in Figure 32, the surrounding volume grid is a C-grid with overset tip and root caps. The background grid is a series of overset cartesian grids of increasing cell size with distance from the rotor hub. The entire quadrilateral grid consists of 4,228,319 nodes.

The solver settings utilize 2nd order temporal and spatial accuracy, an Spalart-Allmaras turbulence model, 0.05^o azimuthal step size with zero subiterations, inviscid off-body computations and non-zero artificial viscosity. To asses the relative accuracy of the present solver settings the parameters just described are compared with 4th order spatial accuracy and reduced artificial viscosity results. Figure 33 compares 4^{th} order and 2^{nd} order spatial accuracy computations. Overall, the reduction in the truncation error of the scheme

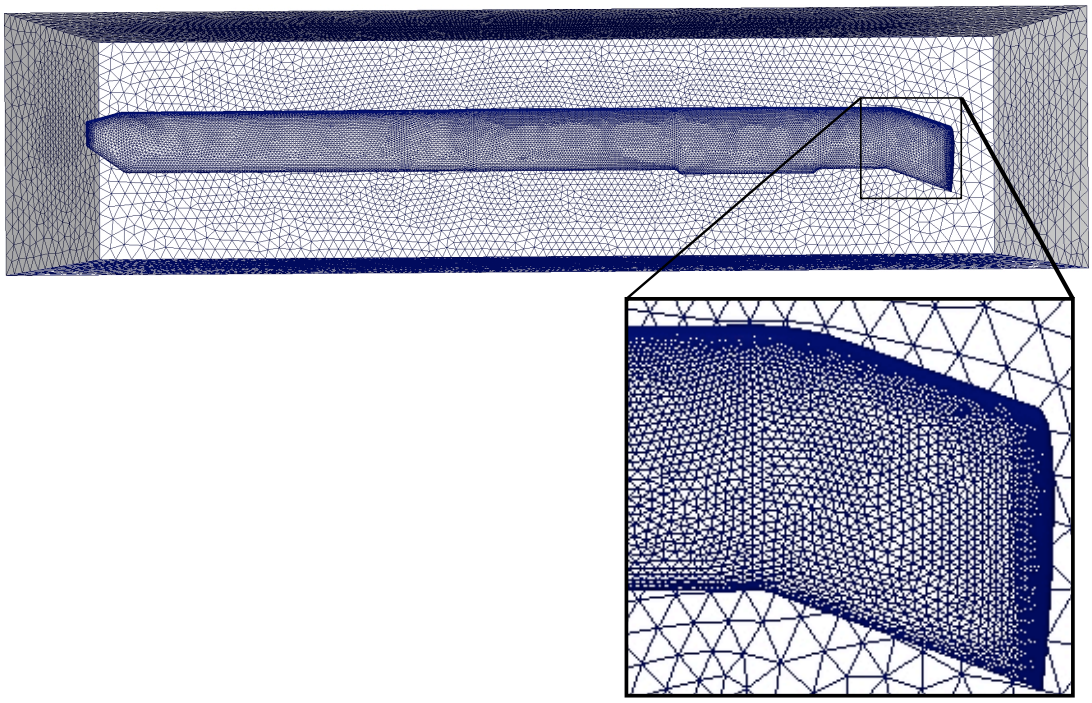


Figure 30: UH60A blade surface grid used in FUN3D

increases the results most on the advancing side of the blade where the gradients are the greatest. Figure 34 compares the results for 4th order spatial accuracy when half the artificial viscosity is applied to the solution. The reduction of the dissipation increases the accuracy in the region of the highest gradients, except at the radial location where sweep begins. Consequently the results for the reduction in the artificial viscosity resemble the results for the reduction to 2nd order spatial accuracy.

4.1.5 Grid Comparison

In addition to differences in the numerical schemes used to compute the results there are several differences between the two grids that may also contribute to differences in the results. The advantage in using an unstructured overset grid instead of a structured grid is that the size of the elements in the background grid can gradually increase in size instead of using an embedded grid pattern seen in structured overset grids. This gradual increase

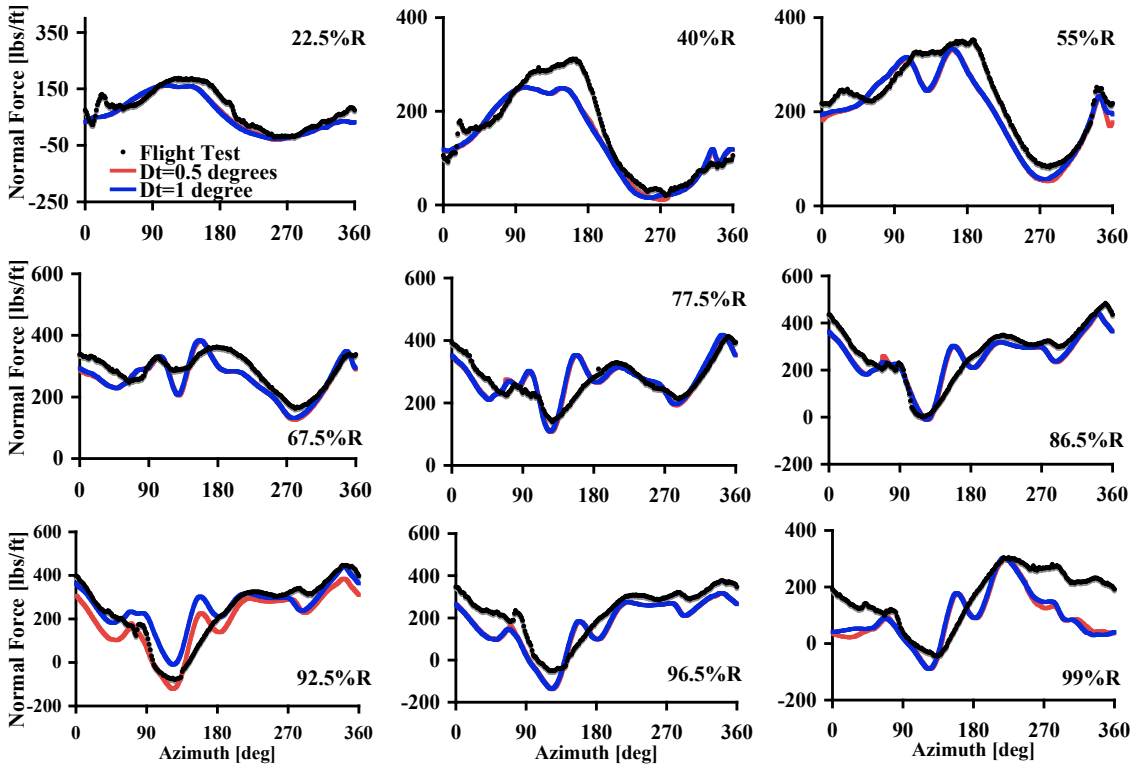


Figure 31: UH60A case 8424 comparison of the normal forces [lbs/ft] for different time step sizes

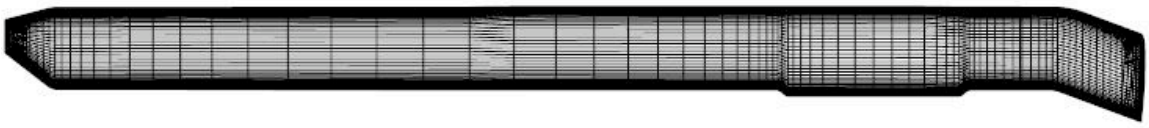


Figure 32: UH60A blade structured grid surface

preserves many of the flow features further from the rotor longer than is seen in a structured grid. Figure 35 illustrates the impact of a sudden increase in grid cell size on the flow field solution. This image shows the vorticity magnitude of a slice of the retreating blade at $r = 62\%R$. As the wake vortices propagate away from the rotor, the flow information is passed to coarser sets of grid points. With fewer grid points to store the profile of the vortices, the resolution of the vortices is lost. This loss of resolution occurs during an abrupt change in the grid cell sizes, as is shown in the structured grid slice. However, the unstructured grid does not show as rapid an increase in cell size, although a loss of

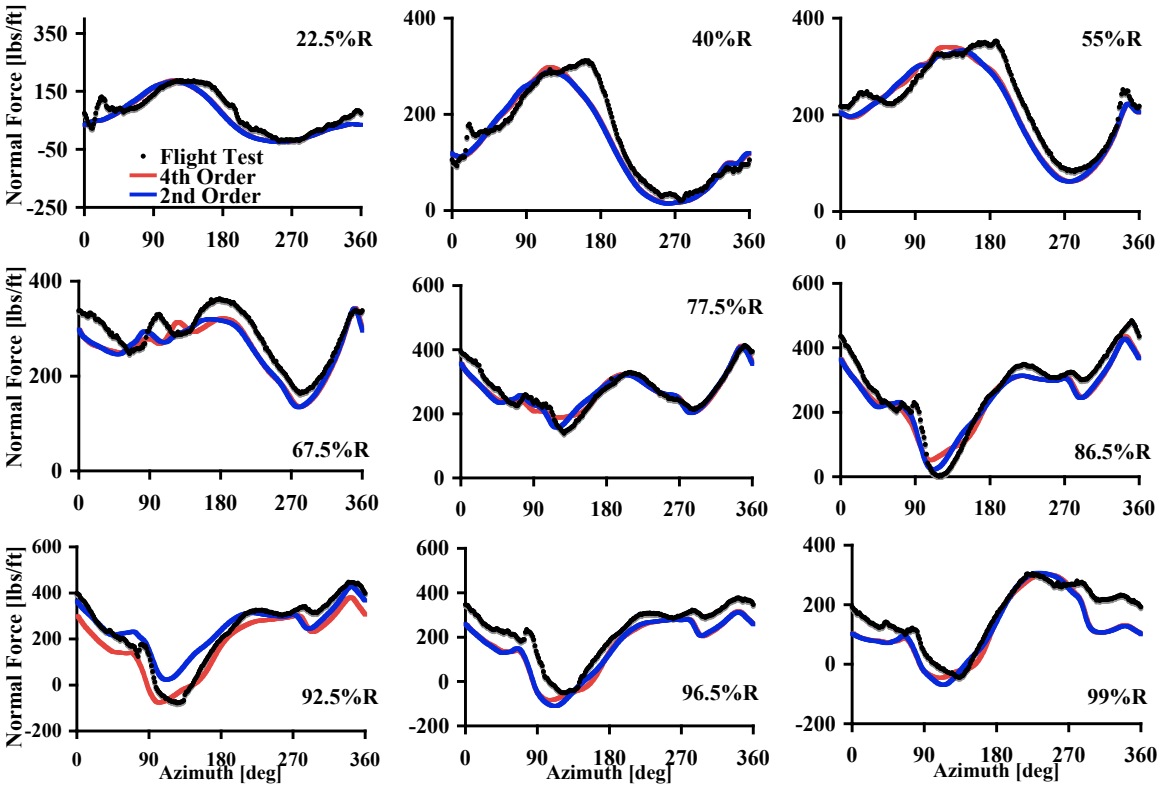


Figure 33: UH60A case 8424 comparison of structured CFD 4th order and 2nd order spatial accuracy results for the normal force [lbs/ft], results courtesy of reference [3]

resolution is seen with increasing grid cell sizes, and thus does not lose resolution as close to the rotor as the structured grid. Other differences between the specific structured and unstructured grids used in this research are the modeling of the tab and the trailing edge. These differences are highlighted in Figures 36 and 37. Although the surface geometries are identical the structured grid has less resolution around the tab transition points, as well as having highly clustered cells behind the trailing edge because of the viscous layering in the C-grid. These differences are highlighted in order to illustrate that although both CFD codes model the Navier-Stokes equations, there are many differences between the two methods that can affect the results.

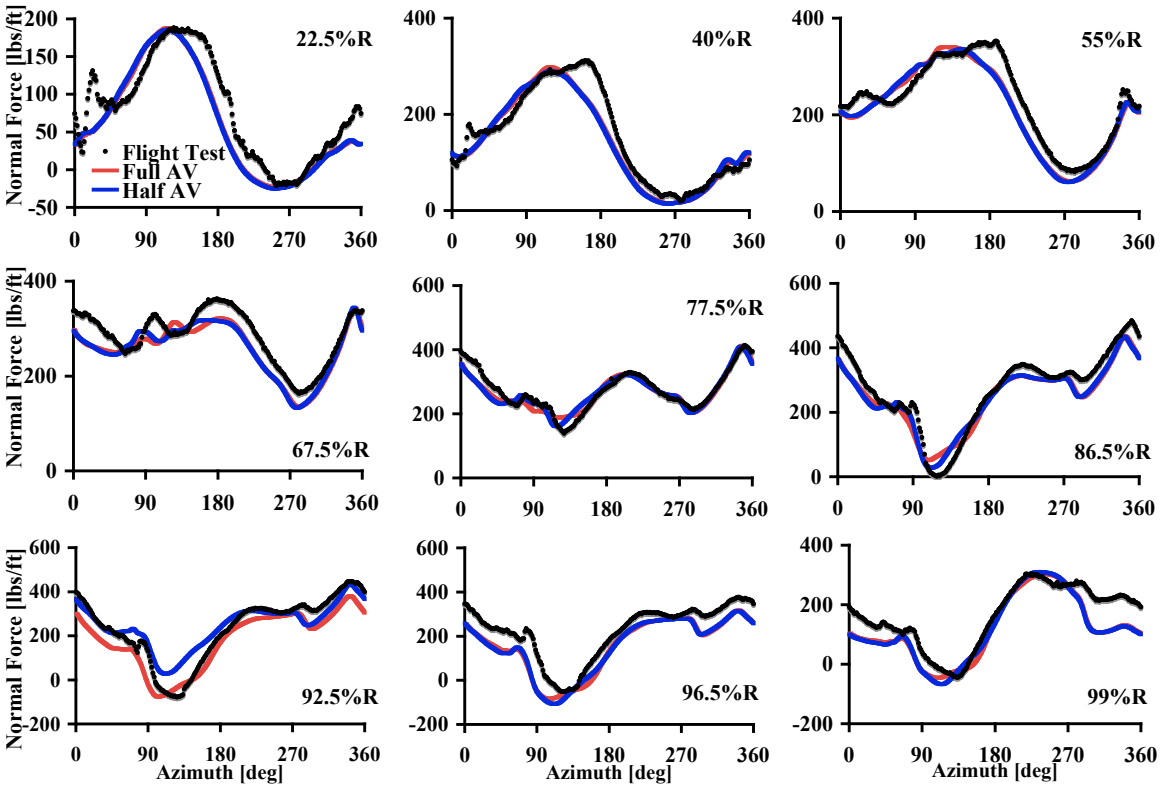


Figure 34: UH60A case 8424 comparison of variations in the artificial viscosity within the structured solver for the normal force[lbs/ft], results courtesy of reference [3]

4.2 GT Rotor-Fuselage

The GT case is based on an experiment performed at Georgia Tech [78, 79, 80] in the John J. Harper 7 ft x 9 ft wind tunnel [81]. The purpose of this test was to obtain data to examine rotor fuselage interaction phenomena. The geometry was simplified so that the number of factors contributing to the flow phenomenon would be as few as possible, and thus the contributions of each remaining factor to the flow field would be easier to identify. The model is a cylindrical fuselage with a hemispherical nose. The system was anchored within the wind tunnel using a strut attached to the rear of the fuselage. The dimensions of this system are indicated in Figure 38. The rotor is a two bladed teetering rotor anchored to the wind tunnel using a shaft tilt of $-6.0°$. The blade cross section is a NACA0015 airfoil with a root cutout of $0.027R$ and a chord length of $0.1881R$. The only motions allowed to the rotor blades are flapping and rotation about the hub.

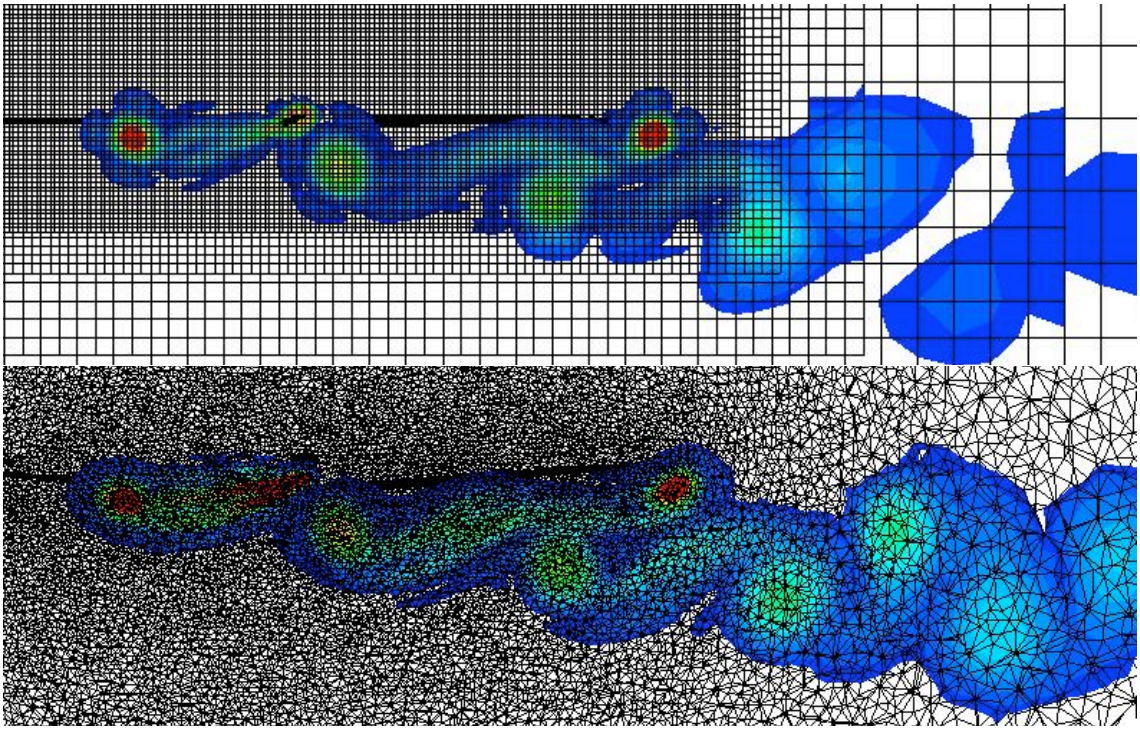
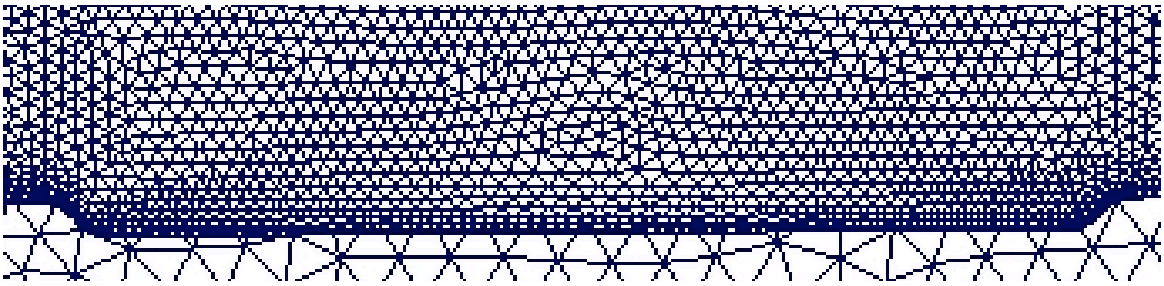
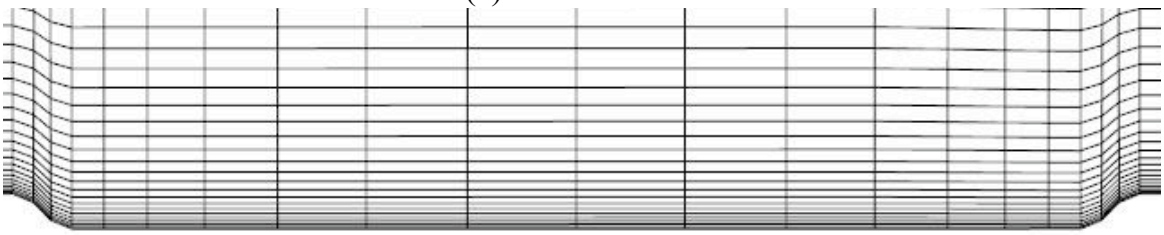


Figure 35: Comparison of the impact of the structured and unstructured grid density on the flow field solution quality, structured data courtesy of reference [3]

For the overset cases the struts and the wind tunnel walls are left out of the final grid in order to directly compare with the data presented in reference [2]. The unstructured grid, in Figure 39, is made up of a background fuselage grid and a blade grid with 1,870,639 nodes and 10,819,705 tetrahedral elements. The blade surfaces are represented in blue inside of the red blade grid. The teetering blades move as a unit, thus the motion algorithm only needs to move one grid in order to apply the motion to both blades. The image of this grid shows the blade grids cutting into the fuselage surface. This is not a problem since the hole cutting algorithm will blank the necessary nodes so that the computations in this region are not performed when the blade grid overlaps the fuselage. The blade pitch is built into the grid using a fixed angle of 10.0°, and the flap motion is controlled by the equation $\beta = -1.94\cos(\psi) - 2.02\sin(\psi)$. This case is therefore ideally suited for a validation case, since the original motion algorithm was able to model all of these motions [2]. Therefore, application of the new motion algorithm should yield identical results. The

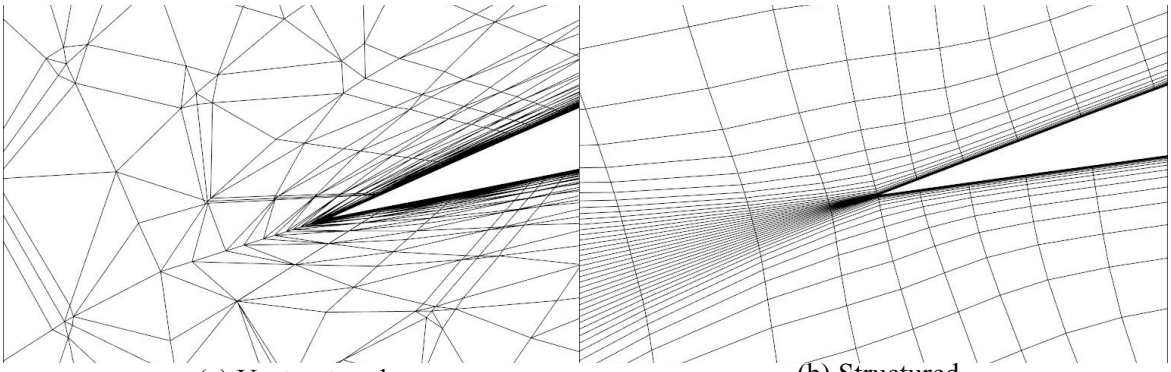


(a) Unstructured



(b) Structured

Figure 36: UH60A blade tab comparison of structured and unstructured grid surfaces



(a) Unstructured (b) Structured

Figure 37: UH60A blade trailing edge comparison of structured and unstructured grid slices

flight conditions for this case are shown in Table 4. The freestream Mach number is well below the compressible boundary, and the rotor tip speed is just below this boundary. Thus, this model may also be used to test the low Mach number preconditioning algorithm.

4.3 ROBIN Rotor-Fuselage

The ROBIN (ROtor Body INteraction) case was tested in the NASA Langley 14 ft by 22 ft wind tunnel [82]. The purpose of this case is to study rotor fuselage interactions. However, unlike the GT case described previously, the complexity of this system is closer to

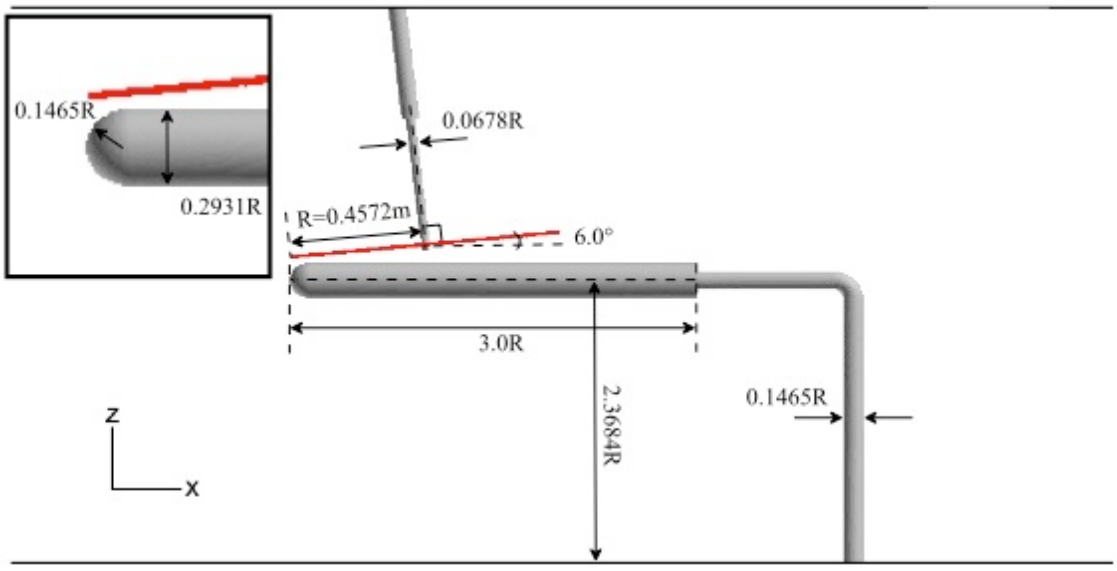


Figure 38: GT grid surface dimensions

what would be seen in a real helicopter. This test was run using three different configurations, Freeman and Mineck [83], Ghee and Elliott [84], and Mineck and Althoff [85]. The configuration that will be considered here is the Mineck and Althoff experiment and is illustrated in Figure 40. The fuselage geometry is represented by a set of functions which may be used to compute the locations of points on the fuselage surface. The ROBIN helicopter is a four bladed model, using a constant NACA0012 airfoil section and linear twist of $-8.0°$. The chord length of the airfoil sections is $0.0663L_{ref}$, where L_{ref} is $3.279ft$. The

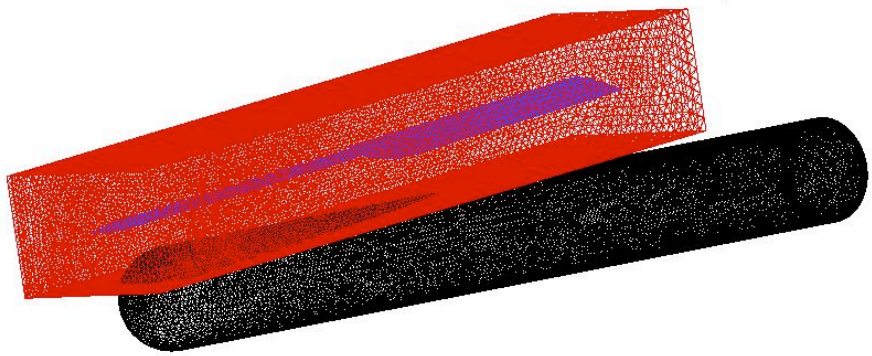


Figure 39: GT overset grid

Table 4: GT case flight conditions

μ	0.1
C_T	0.009045
M_∞	0.295
M_t	0.0295
α_s [deg]	-6.0

results from this case further validate the results of the rigid motion algorithm, as well as showing the influence of small flap hinge offsets and lead-lag motion as compared to the less articulated case. The dimensions of this grid are given in Figure 40. This case has all types of articulation, including pitch, lead-lag, and flap. Also present are shaft tilt and hinge offsets. This model is a four bladed model with full articulation. The motions present are pitch, flap, and lead-lag, although the lead-lag motion provides only a small constant contribution. There are hinge offsets in this model, which the previous partially-articulated algorithm was unable to model.

The unstructured grid consists of a background grid, containing the fuselage and struts, and four overset blade grids. The background grid contains 1.7 million nodes, and each blade grid has 250,000 nodes. This configuration is shown in Figure 13. Unlike the GT case, each blade is contained in its own grid. This is necessary since each blade moves independently of the other blades. Having multiple overset blades to control adds complexity to the solution of the rigid motion equations, and is used as a further validation of this new algorithm.

The flight conditions for this case are given in Table 5. Unlike the previous case, this case does not lie entirely below the compressible boundary. The fuselage is located in low speed flow, but the rotational speed of the blades creates a condition of high speed compressible flow around the rotor. The result is that it would not be effective to run the case using only the incompressible equations; the flow around the blades would not be properly modeled in this case. Compressible equations or the addition of low Mach

number preconditioning would be the best choices. The blade control angles are controlled by the expressions, $\theta = 16.3° - 2.7° \cos(\psi) - 2.4° \sin(\psi)$, $\beta = 1.5°$, and $\delta = 0.95°$. The flap, lag, and pitch hinge offsets are located at $r = 5.9\%R$. The addition of the hinge offsets and lead-lag change the resulting motion by a small amount as compared to the partially articulated case

Table 5: ROBIN case flight conditions

μ	0.151
Ω $[rad/sec]$	209.44
M_∞	0.08
M_t	0.529
C_T	0.00643

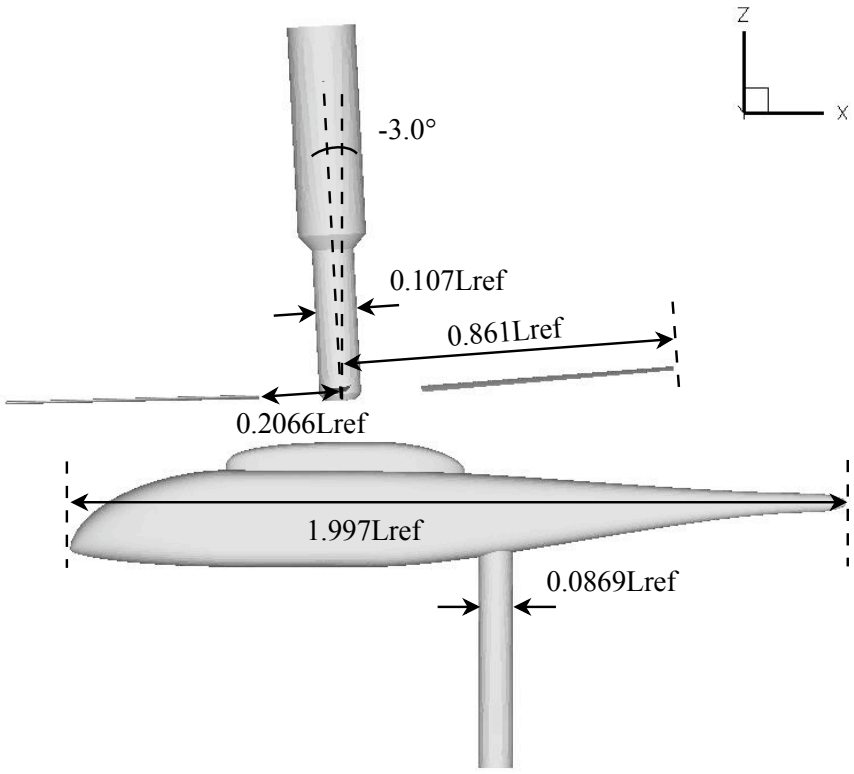


Figure 40: ROBIN grid surface dimensions

CHAPTER 5

RESULTS: CSD-FW STUDIES

5.1 Computational Timing Analysis

The results for the timing improvements include both the improvements after modifications were made to the serial code, and also after the code parallelization. As mentioned in section 3.2, small efficiency improvements to the code can have large overall computational savings. The main reason for this improvement can be seen in the structure of the most expensive part of the code. The application of the Biot-Savart law to the wake vortices is by far the most time consuming algorithm. The seven layered nested loop structure seen in these computations means that a small change to only one line of the code inside these loops has an additive effect on the total time savings. Other parts of the code have similar, but smaller, loop structures and thus this argument extends to many other parts of the code as well. Shown here are the results of these code changes to the computation time. These changes show the timing after modifications to the code have been made, with a comparison to the newly developed parallel version of the code.

5.1.1 Free Wake Coding Improvements

The first step in improving the efficiency of the code is to modify the original algorithm using techniques to clean, and streamline the code. The times represented in Figure 41 compare the use of MFW before and after the improvements inside of the CSD-FW coupling. These computations show the time per iteration with file passing being performed every one revolution on a Linux Evolocity II running 3.6 GHz processors. Since no modifications were made to any part of the code other than the free wake code, all efficiency gains are from the free wake computations. For only one or two wake trailers there is almost no

change seen in the run time. However, as the number of wake trailers is increased, the run time increases exponentially. This directly ties into the nested loop argument because only one or two wake trailers significantly reduce the number of computations performed by the interior loops. After the improvements to the code the same type of exponential increase is seen in the run time, but this growth is slower. This is because of the additive nature of the improvements. More wake trailers mean that the code will remain in the loop structure longer and will thus compute the same lines of code more often. Improving these repeated lines of code will have a greater impact the longer the loop is run. Thus, the computational savings are greatest for larger numbers of wake trailers. Porting the code to a MacBook Pro with a 2.33 GHz Intel Core 2 Duo processor yields a further 1.3 times speed improvement over the Linux machine. The slower time on the faster machine is attributed to shared memory competition among all the jobs running on the same node.

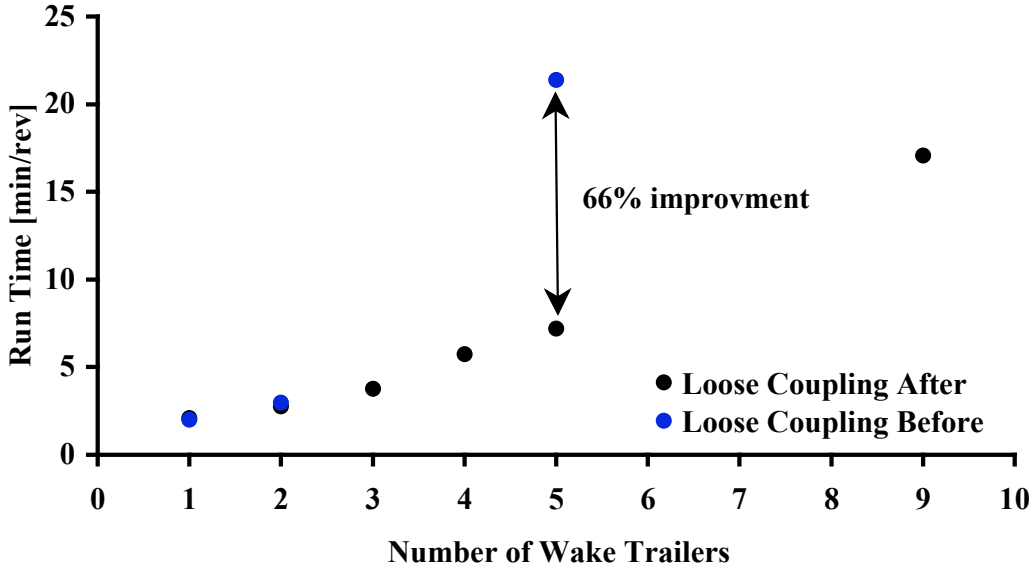


Figure 41: Run time on a Linux Evolocity II for different model resolutions and coupling schemes before and after code efficiency improvements are made to the free wake code

5.1.2 Free Wake Parallelization

For the parallel code, the goal is not to merely improve the run time, but to eliminate the quadratic increase of the run time with increasing model resolution. The parallel version of the code succeeds in this goal. Figure 42 shows the timing results of the parallelized code. The quadratic increase of the run time has been eliminated, making the code more efficient for larger numbers of wake trailers. The run time of the code now increases linearly as the number of wake trailers is increased. This linear increase occurs because not all loops in the code have been parallelized, only the ones which present the greatest computational expense. So there is still some dependence of the code on model resolution, and thus the time will still increase. However, the elimination of the quadratic increase is the best possible result, since parellelizing all loops in the code will begin to introduce MPI overhead to the computations.

This linear increase is a major improvement over the serial version of the code. Ideally, only one wake trailer per processor will run the fastest, but if not enough processors are available, more wake trailers can be assigned to each processor available. The consequence of this coarser partitioning of the model is to increase the slope of the run time with increasing model resolution. However, the exponential increase is not reintroduced for smaller numbers of wake trailers per processor. Thus the free wake code can be run much more efficiently for larger numbers of wake trailers.

5.2 Parameter Study

For the data presented in this section all results use the UH60A rotor, case 8534. This section shows the effect of different aspects of the model on the results, and performs an evaluation of the different methods. These include, but are not limited to, the various methods used to pass data, the types of data, as well as an analysis of the impact of using trim, and changes to the aerodynamic model.

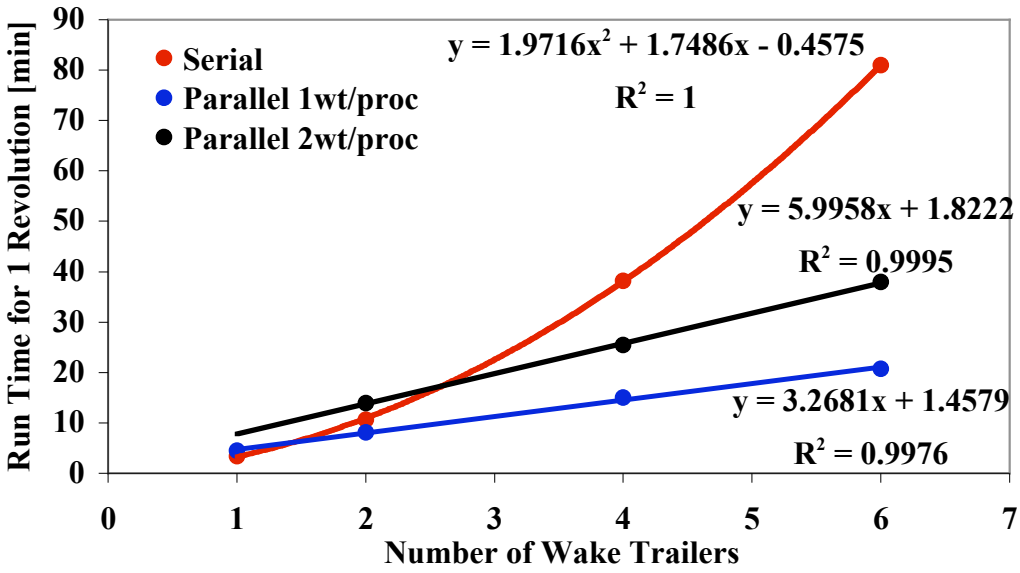


Figure 42: Parallel run time for different numbers of wake trailers per processor using an IBM P4 with 1.7 GHz processors

5.2.1 Comparison of Loose and Tight Coupling

The two types of file passing methods differ in timing but not in theory. Therefore, the only difference between these two methods, for a steady level forward flight case, is the run time. In order to directly compare loose and tight coupling methods, consider a case where the trim feature of DYMORE has been turned off, so that the influence of the trim computations is not present in the solutions. A sweep of free wake azimuthal resolutions is then computed and compared with a tightly coupled run. Comparing the results in Figure 43 illustrates that for the case where a 1.25^o step size is used for both the tight and loose cases the results are identical. For coarser loosely coupled model resolutions of greater than 1.25^o there is a very small difference in the region where the vibratory loads are the strongest, and thus where the wake will have the most influence. Although the solutions are identical, there is a large difference in the time needed to compute each of these types of solutions. It takes 85% more time to compute using tight coupling than using loose coupling. Based on these results, the comparisons made here, and in later chapters, rely on

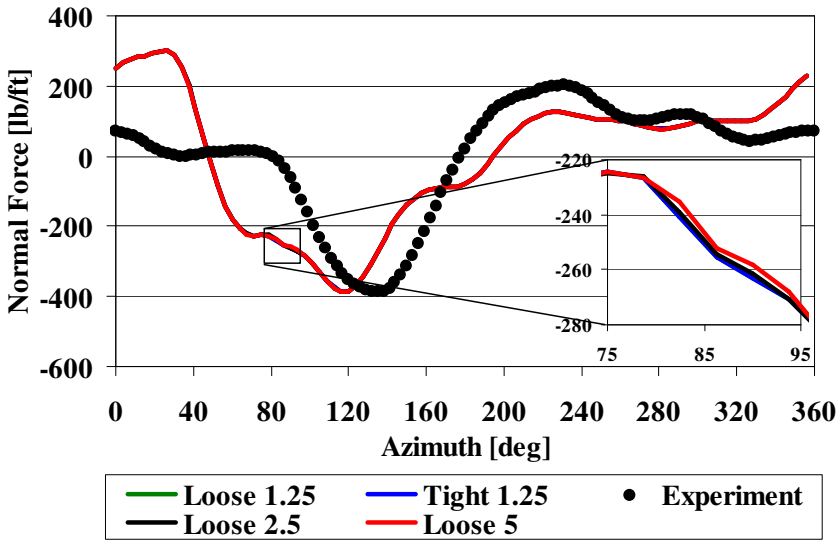


Figure 43: UH60A comparison of loose and tight coupling for CSD-FW coupling

solutions obtained using loose coupling instead of tight coupling.

5.2.2 Trim Analysis

The purpose of trimming the solution is to change the blade controls until target hub loads are predicted. This process changes the blade pitch motion until the thrust, pitching moment, and rolling moment at the hub match these target loads. The target loads themselves represent the forces and moments of the fuselage on the rotor, which in a physical system that has both rotor and fuselage in hover are balanced such that the total forces on the system are equal to zero. Trimming the solution alters the magnitude of the final prediction, but has little effect on the higher harmonic characteristics of the results, as illustrated in Figure 44. This small effect on the higher harmonics is because the trimmer only has control of the first three coefficients of the Fourier series used to compute the blade pitch. Thus the trim uses only the zeroth and first harmonics, where the zeroth harmonic represents the collective pitch, or thrust control, and the first harmonic sine and cosine coefficients control the rolling and pitching moments. Therefore, the trimmer has influence on the lower harmonics of the aerodynamic computations only. This analysis is limited to the range of angles of attack that the simplified aerodynamic model is able to predict accurately.

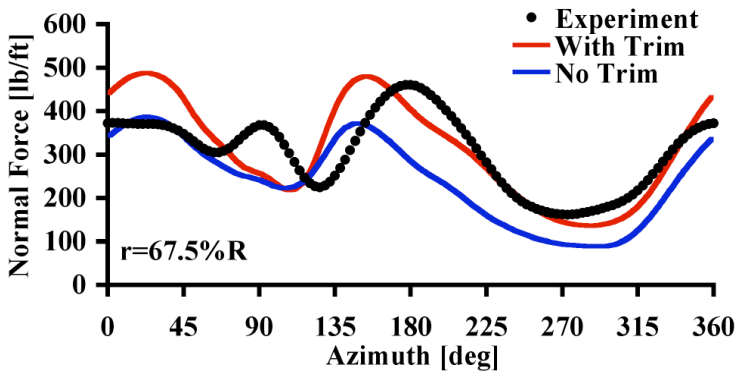


Figure 44: UH60A comparison of the normal force [lbs/ft] with and without adding trim to the analysis

The trimming is computed at each time step to match defined hub forces and moments. The collective and first cyclic of the pitch function are adjusted until the trim targets have been reached. The convergence time of this trimmer is a function of the most sensitive computation. In this case, the thrust is the least sensitive parameter and thus is the fastest to converge. The two moment targets take longer to trim than the thrust target as plotted in Figure 45. This is because the moment computations are more sensitive to variations in the structural and aerodynamic models than the force computations. The trim convergence in Figure 45 represents 130 revolutions of data. For most cases 60 revolutions is found to be more than enough to completely converge the solution, which is equivalent to a simulation time of 14 seconds, and is set as the default number of revolutions for the results presented.

Modifications to the other trimmer settings do not have any affect on the solution; these affects have only been examined to identify the ideal solution convergence. Different modifications that can be made include, altering the magnitude of the perturbation changes, altering the gain, and altering the method by which a Jacobian is obtained. Of these changes, only the gain settings have some impact on the convergence, as plotted in Figure 46, the other settings must be selected such that the computation of the Jacobian is successful, but they have no impact on the solution. The gain adjusts the magnitude of the step in the control angles applied at each time step, and thus directly affects how fast the loads change.

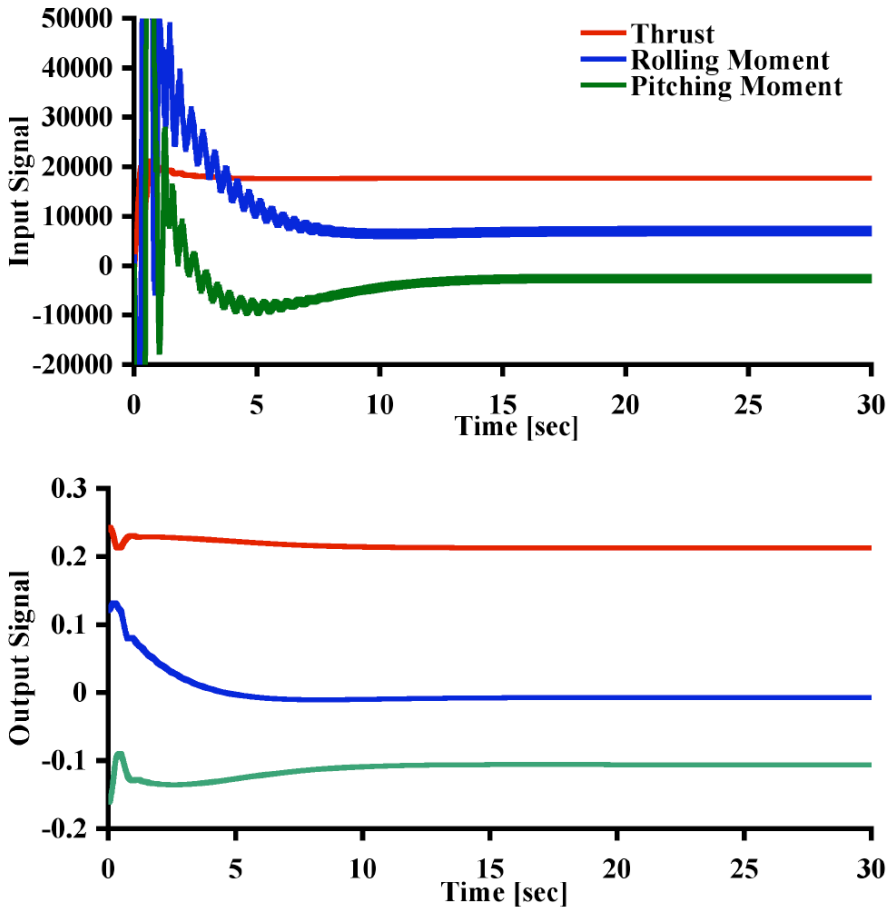


Figure 45: UH60A trim convergence for an example run, where the input signals are the total hub forces [lbs] and moments [ft lbs] and the output signals are the collective and cyclic pitch inputs [rad]

This setting has been selected to be 0.6, since this setting is low enough not to cause the solution to overshoot the target.

5.2.3 Model Modifications

During the development of the CSD-FW coupling, changes to the computational models were made to study the impact of the changes on the stability of the system. Although the simplifications presented below are no longer present in the final coupling, the results of the analysis lead to some interesting conclusions, and have been compiled below.

Initially, the structural data passed from the CSD code to the free wake code was not

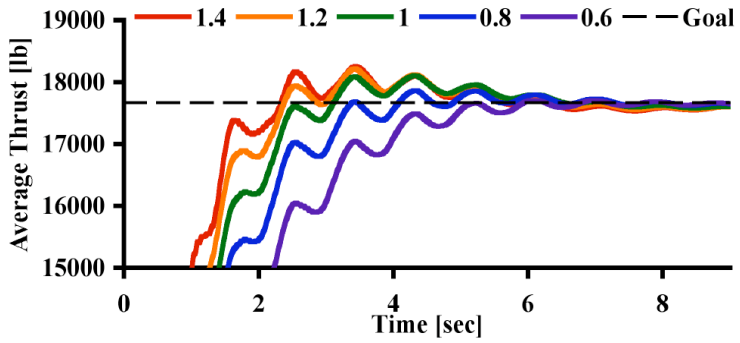


Figure 46: UH60A thrust convergence for different auto pilot gain settings

completely used by the free wake code. The lead-lag elastic blade deflections were neglected in initial computations. This lead-lag motion was later included in the free wake code, and was found to have a very small effect on the solution when compared to the experimental data. The true impact of the lead-lag hinge is to eliminate the blade root bending moment along the vertical axis, but since this motion is always included in the structural model, the computation of this moment has not changed. Including the lead-lag motion in the free wake code only serves to change the effective azimuthal locations of the blade attached vortex filaments, and thus the boundary conditions of the free wake computations. However, the change in this motion is very small and has little impact on the wake geometry, which has only a small impact on the aerodynamic loading. The largest difference occurs on the advancing side of the blade where a small favorable phase shift is induced. This indicates that the largest lead angles can be found on the advancing side of the rotor. Although this effect is small, since the inclusion of this angle makes the analysis more complete, it has been left in for all remaining cases.

The effect of changing the aerodynamic twist distribution in DYMORE is to adjust the level of the solution not the shape or the phase, as shown in Figure 48. The purpose behind investigating the linearization of the aerodynamic twist is that during the implementation of the loads coupling method it was found that the increasing instabilities were also affected by the sudden change in twist along the blade. In an attempt to stabilize the code, a

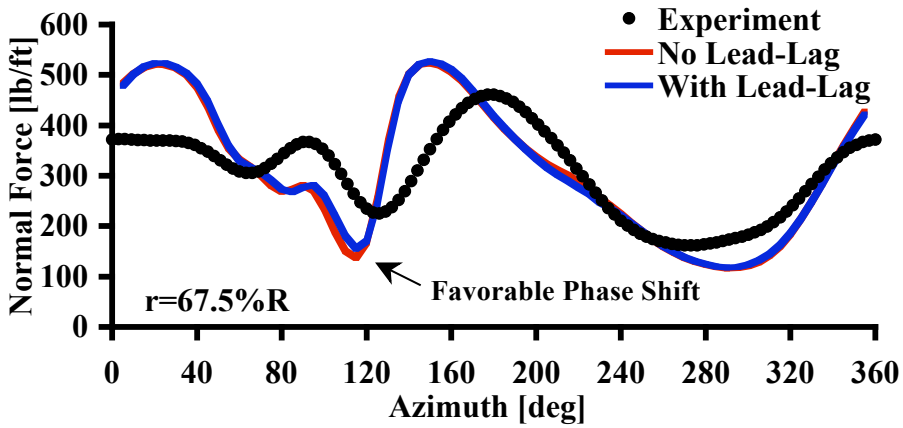


Figure 47: UH60A comparison of the normal force [lbs/ft] with and without adding lead-lag to the analysis

linear twist distribution was applied so that there were no changes in the slope of the twist function anywhere along the blade. The changes to the twist were later determined to be unnecessary, but the results have proven to be useful. In the region close to the root of the blade at 40%R, there is only a small change if the twist associated with the blade tab and tip is removed. In this region the local twist remains the same, thus the aerodynamic computations are using the same local twist that was being used before. However, in the tab and tip regions shown in the remaining figures, the results change more dramatically if the linear twist distribution is used. The reasons for this change are that changing the twist of the blade alters the angle of the blade relative to the local impinging flow and thus alters the computations of the aerodynamic loads. Since the aerodynamic model applied is a 2D model, changing the twist in isolated regions will have the largest impact on only those regions. However, the loading on the sections with unmodified twist will still be affected, because the trimmer will have to change the control angle applied to the entire blade in order to reach the same trim level.

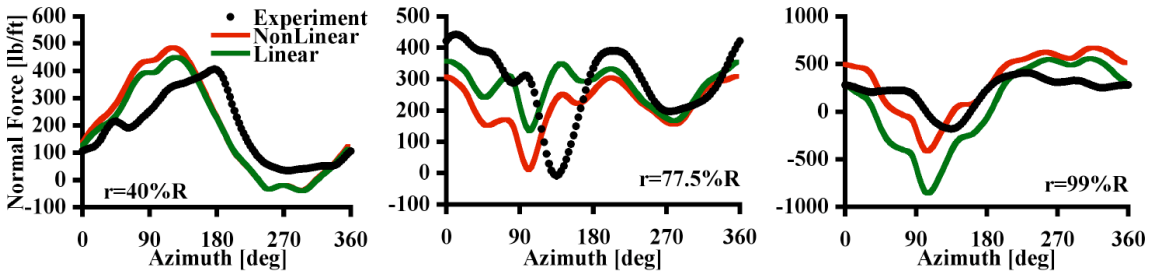


Figure 48: UH60A twist distribution effects on the normal force [lbs/ft]

5.2.4 Aerodynamic Model Variables

Changing the constant aerodynamic coefficients used in the DYMORE Peters aerodynamic model has some impact on the results. The two sets of coefficients tried are given in Table 6. The set 1 data are the original sc1095 settings and the set 2 values were selected from a compilation of experimental and computational data given in reference [86]. The differences show that there is a small improvement in the results using set 2 instead of set 1, and thus set 2 may more accurately reflect the actual coefficients of the sc1095 airfoil. However, these constant coefficients lack much of the high speed and high angle of attack information seen in the real system. A more accurate method of modeling the airfoil is to use the option of including static experimental data to replace these constant coefficients. The data are stored in tables of lift, drag, and pitching moment for a range over angle of attack and Mach number. The aerodynamic computations extract the current steady aerodynamic loads from these tables and incorporates them into the unsteady computations as described earlier in the methodology chapter. This method includes the nonlinear characteristics seen in the static airfoil behavior, and thus yields more accurate results and is the preferred method used for this research. However, as illustrated in Figure 50, these benefits are seen more in the computation of the pitching moment than in the normal force. The addition of the nonlinear airfoil effects has a greater effect on the pitching moment, because the normal force behavior is approximately linear for angles of attack below stall and thus the normal force will not change as much when these affects are included.

Table 6: UH60A DYMORE model constant aerodynamic coefficients

Set	1	2
Cl	5.73	6.8755
Cd	0.018	0.018
Cm	0.0	-0.01

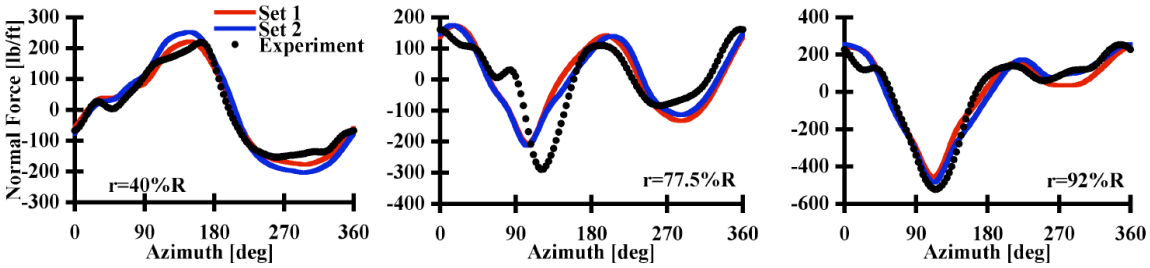


Figure 49: UH60A comparison of 1-10p normal force [lbs/ft] results for different sets of aerodynamic coefficients where set 1 and set 2 are defined in Table 6

5.2.5 Data Passing Types

Various levels of coupling have been applied in order to evaluate the effectiveness of different components within the schemes. Three types of coupling are presented here, angle of attack, structural, and frozen. Comparing the different type of data passing for the FSI coupling shows that the best results are obtained by using the closest coupling. Figure 51 compares the frozen, structural, and angle of attack coupling methods.

The frozen method has the smallest degree of coupling, because not only is no aerodynamic data being passed between the structural and free wake codes, but the aerodynamic computations inside of the free wake code have also been frozen to whatever they were at the first revolution. The only changes present in the free wake computations are for the wake geometry, not the wake vortex filament strength. This is the least coupled method because the vortex strength of the physical wake is not being properly modeled, since the blade circulations have been frozen. Thus, the wake used to represent the inflow distribution on the rotor plane is not as accurate as one in which the circulations are variable, even if the wake geometry is properly modeled by using the more accurate boundary conditions

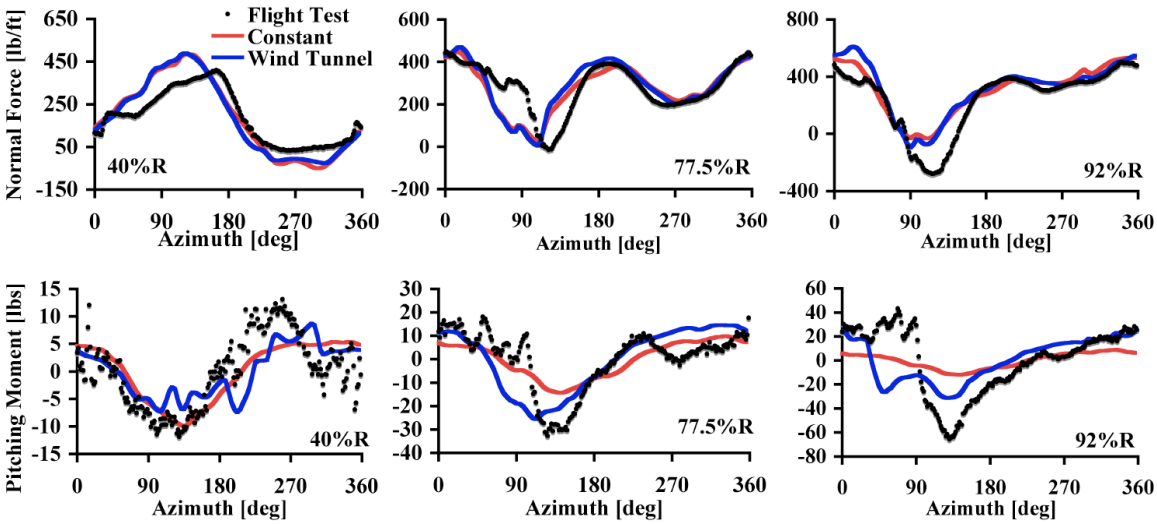


Figure 50: UH60A normal force [lbs/ft] and mean removed pitching moment [lbs] results using wind tunnel static data tables versus set 2 constant aerodynamic coefficients

from the structural coupling. The solution of the rotor wake equations is different because of the different induced velocities from using a constant circulation distribution. Thus, the wake geometry is affected by the circulation distribution.

The structural method has a closer coupling than the frozen method. This method passes only the structural deflections, but does not freeze the circulation computations in the free wake code. Therefore, the same structural data used to compute the wake geometry is passed to the free wake code, but the strengths are not dependent on the actual blade circulations, but on circulations computed using a method which runs independently of the aerodynamic loads computations. This means that although the wake geometry is correctly computed, the wake vortex strengths are only approximated, because they are decoupled from the blade circulations computed in the CSD code. This method has mixed results, but is closer in magnitude to the flight test data than the frozen method, and contains more higher harmonic data.

The closest coupling is obtained by passing both the structural deflections and the effective angles of attack, which are used to compute the blade circulation in the free wake

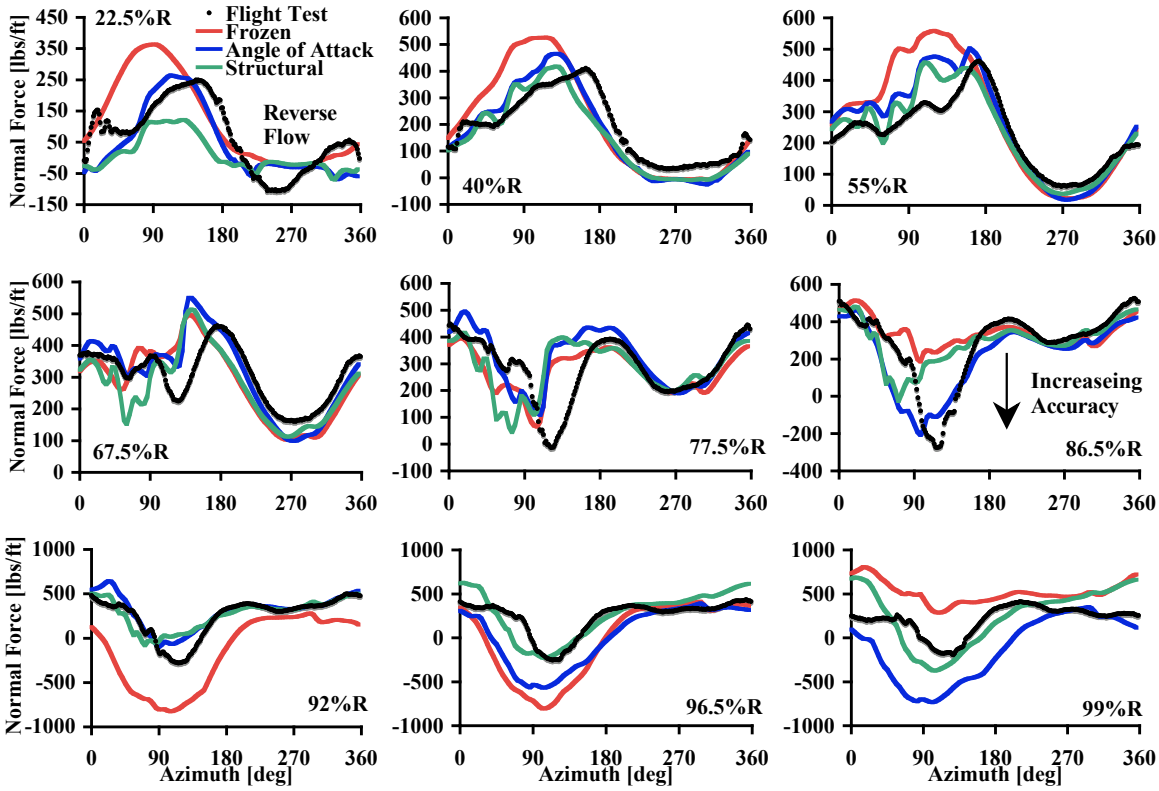


Figure 51: UH60A comparison of different data passing coupling methods

code. The angle of attack method has the best overall match to the experimental data. This method passes the blade deflections so that the free wake code has the correct boundary conditions when computing the wake geometry. The effective angle of attack is passed so that the blade circulation, which is shed into the wake, has the correct wake vortex strength. This method is dependent on variables computed by the same aerodynamic model that is used to trim the computations. This fully-coupled method uses the fewest assumptions regarding the features of the wake. It does not assume that the wake vortices are of constant strength, like the frozen method, and it does not assume that the vortex strengths are independent of the blade aerodynamic loading. Neither of these assumptions accurately represent the physical system in which the wake vortex strength is a function of the aerodynamic loading on the blade. Thus the angle of attack method is the default method for which the remaining cases will be run.

5.2.6 Independent Code Evaluation

The overall effectiveness of the coupling method as compared to the uncoupled codes (Figure 52) shows that the coupling does better than either of the independently run codes. The rigid trimmed results from MFW miss many of the major characteristics, even on the retreating side of the rotor, as well as predicting high fluctuations in the reverse flow region. The aerodynamic model used here is a variation of the Lieshman-Beddoes model, and is different from the Peters model used in the DYMORE computations. However, the present results indicate that the assumptions made by the aerodynamic model do not include any of the higher harmonic data. This is because, as a default option, MFW turns off the unsteady part of the aerodynamic computations when the trimmer is being used. These unsteady aerodynamic elements are necessary to accurately predict the vibratory loading on the rotor, and without them only lower harmonic steady results are being predicted. When DYMORE is run independently using the Peters-He dynamic inflow model, run with 231 states, the results are smoothed out, except in the tip region where it appears that the dynamic inflow model has trouble with the higher speeds. The use of the dynamic inflow model provides a time-averaged inflow distribution, and does not accurately represent the time-accurate nature of the rotor wake. In the physical system, as the blades rotate, the helical wake structure propagates further into the flow field, as these wake vortices move and dissipate over time, their influence on the rotor plane changes continuously. The time-averaged prediction of the dynamic inflow model does not model these time dependent effects, resulting in the smoothed out solution seen in Figure 52. However, the aerodynamic model still has unsteady computations, so some of the higher harmonic vibratory load predictions are still being computed. The coupled results are similar in character to the DYMORE alone results, but with the addition of higher harmonic content especially on the advancing region where unsteady aerodynamics are more important. This is because the coupled solution uses both the time-accurate inflow computations as well as the unsteady aerodynamic load computations, reducing the assumptions made by each of the individual codes. This creates

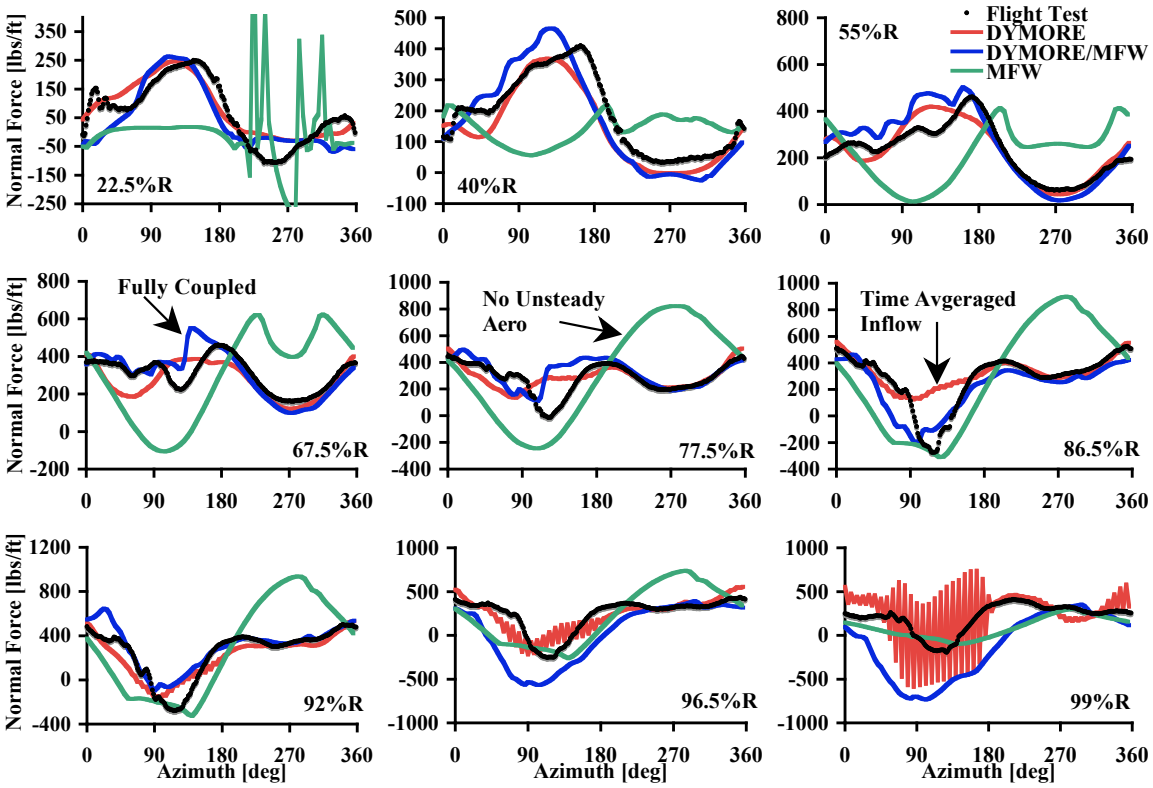
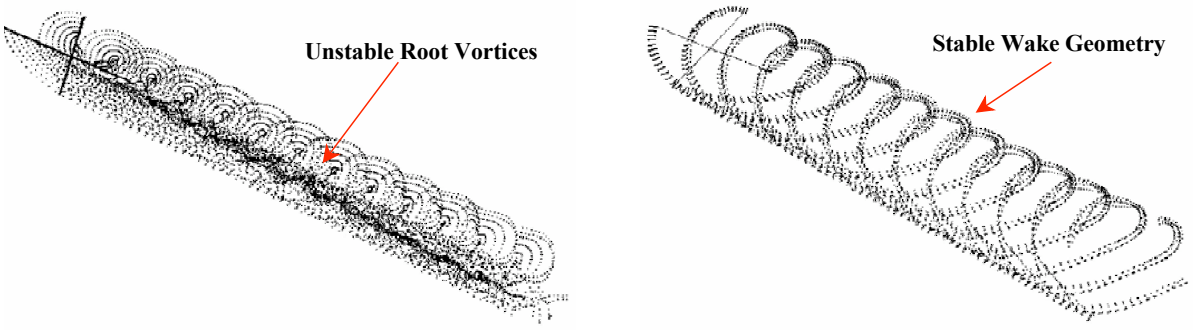


Figure 52: UH60A comparison of CSD-FW coupling with independent DYMORE and MFW results

a more physically accurate prediction, which includes both the time dependent wake effects and the unsteady airfoil motions.

5.2.7 Wake Trailer Analysis

Changing the number of wake trailers does have an impact on the solution, this impact can be described by looking at the number and distribution of the wake trailers. There are many ways in which the wake trailers can be distributed. All cases used in this comparison are computed using the angle of attack data passing scheme. For the evenly spaced distribution, trailers located at the root of the blade generate instabilities that cause the free wake code to crash, as illustrated in Figure 53a. Roll-up of the root vortices is physically accurate, especially when there is no hub present, but the nonphysical features created in the inflow distribution on the rotor plane are not physically accurate. Clustering the trailers at the tip

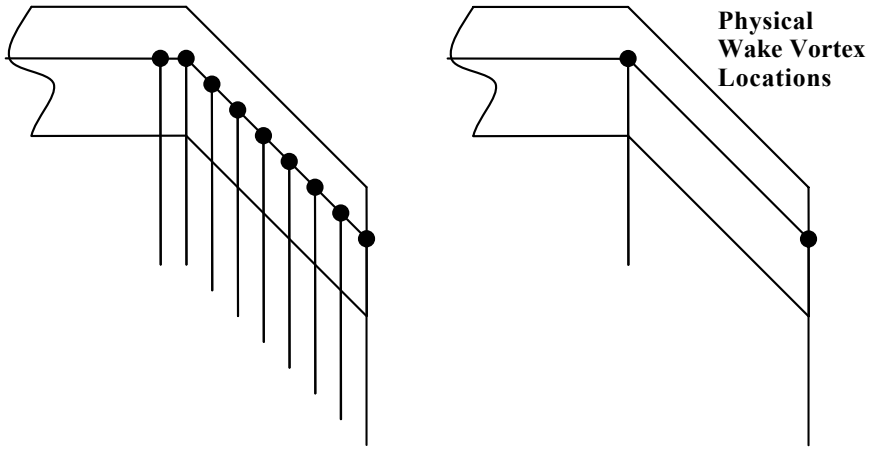


(a) Evenly Spaced **(b) Clustered**

Figure 53: UH60A free wake geometry for different wake trailer distribution types

eliminates the rotor plane instabilities, and improves the results gradually as the number of wake trailers is increased, as illustrated in Figure 53b. The wake geometries shown in this image look as if there is no wake roll-up present in the geometry. However, in Figure 65, the perspective with which the geometry is viewed can mask this effect. Roll-up is present in these images, but it is the perspective that makes it look as if there is no roll-up.

For the angle of attack coupling scheme this improvement with increasing wake trailers is not a function of the number of wake trailers, but of where these trailers are positioned along the blade. Two types of distributions, clustered and user defined, are illustrated in Figure 54. These distributions will be used to compare the differences between wake trailer location and number. The results of Figure 55 show that a nine wake trailer clustered run is identical to a two wake trailer run, where one trailer is at the tip and one trailer is at the inboard sweep location. This clustered comparison also emphasizes the lack of sensitivity to increasing the number of wake trailers in the model. The physical wake in this region of interest has both a tip vortex as well as a vortex which is shed at the sweep transition location. The nine wake trailer case covers both of these locations, with additional wake trailers along the swept region. The two wake trailer case also covers both of these regions, but eliminates all of the extra wake trailers in between. Thus the wake geometry which has the greatest impact on the solution is not one which computes a sheet of wake trailers,



(a) Clustered **(b) User Defined**

Figure 54: Illustration of free wake trailer distribution types

but one which resembles the dominant vortices of the physical wake. The comparison between the one wake trailer case in the same figure shows that these identical results are not obtained for a case which does not model the sweep transition location. Thus, as an extension to this analysis, the wake trailer distribution with the greatest impact on the solution is a four wake trailer model where the trailers are located at the tip, sweep and tab transition locations. The results of this case are shown in Figure 56. This distribution represents the two dominant wake vortices in the flow, as well as any effects on the wake which come from the tab transitions. The tab transition locations have a smaller radial change in the geometry than the sweep transition, but some small vortices will still be generated in this region. The four wake trailer case does show some improvement in the unsteady fluctuations, but it does not have a large impact on the magnitude of the lower harmonic content. Since the main effect of the free wake code is to better predict the unsteady airloads it is not expected that the free wake model will have a large impact on the lower harmonic magnitudes. Primarily the aerodynamic model predicts these lower harmonics. This effect is only confirmed further by this comparison.

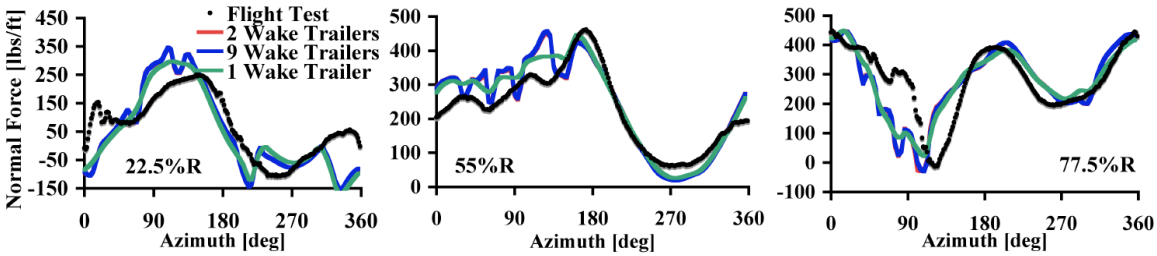


Figure 55: UH60A comparison of a nine tip clustered wake trailer case, a user defined two wake trailer case, and a one tip wake trailer case using constant aerodynamic coefficients in DYMORE

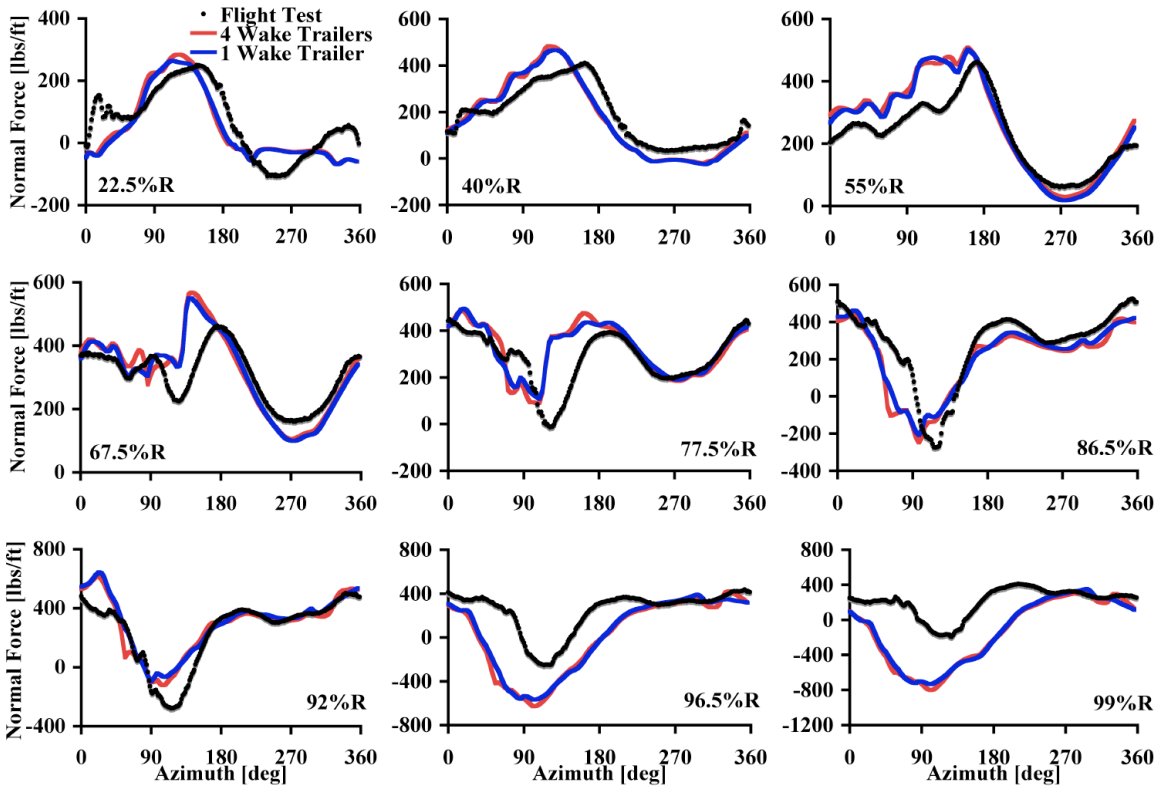


Figure 56: UH60A comparison of a one wake trailer case with a user defined four wake trailer case using aerodynamic tables in DYMORE

5.3 UH60A Case 8534

This high speed case is characterized by high vibratory loading present on the advancing side of the rotor. These vibratory loads appear in the experimental data in Figures 57 and 58. These loads are the result of the higher speeds on the advancing rotor blade. The advancing side from $\psi = 0^o$ to almost $\psi = 180^o$ shows high fluctuations, while the retreating side is much smoother. The predictions are modeled using a one wake trailer free wake grid with Peters aerodynamic theory using a tip loss factor and angle of attack coupling between MFW and DYMORE. The simulation was run to trim convergence to match the target loads given in Table 3. The Peters method does model unsteady aerodynamics, but some of the high fluctuations in the advancing region are not captured. However, the retreating side does not exhibit large unsteady loading and is well captured using simplified aerodynamic theory. The pitching moment results are close to experiment on the retreating side of the rotor near the sweep transition, as well as near the root of the blade outside of the reverse flow region. The results in the reverse flow region, where the angle of attack ranges from 0^o to 180^o, do not reach the peak loading value in either the pitching moment or the normal force. This region is where the rotating velocity of the blades is not able to overcome the forward velocity of the rotor, resulting in a region where the flow is moving from the trailing edge to the leading edge of the blade airfoil sections. As the blade enters this region, the angles of attack change rapidly from close to 0^o to close to 180^o. As the blade moves away from this region it changes back as rapidly from 180^o to 0^o. So even though the blade pitch has not actually been rotated by such a large angle, the flow direction effectively forces this motion. The present aerodynamic model does not predict these high angles of attack and thus it is expected that this region will be missed. This is seen in the $r = 22.5\%R$ plots of normal force and pitching moment where the peaks at about $\psi = 270^o$ are cut off abruptly.

Since the unsteady loading is associated with the higher harmonics, and since the unsteady loads are harder to model than the steady loads, it is expected that discarding the higher harmonics in the experimental and the computed data will yield better results. These

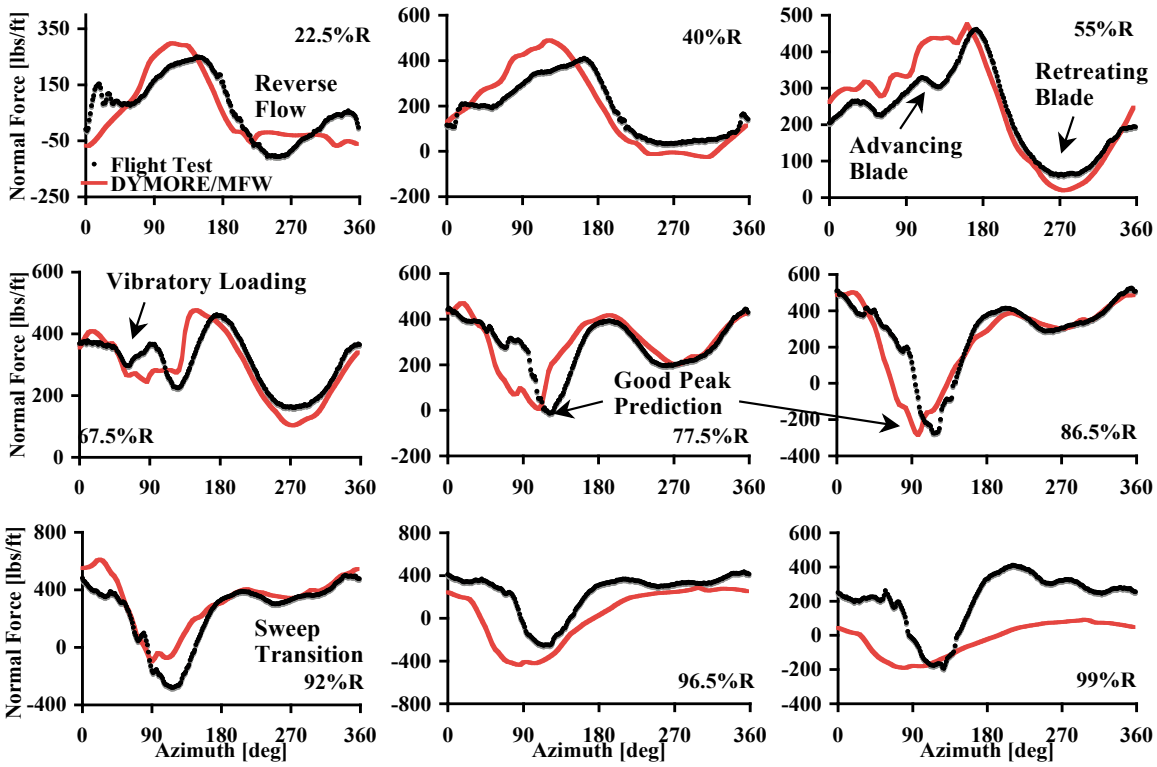


Figure 57: UH60A case 8534 normal force [lbs/ft] using free wake angle of attack coupling

cases are plotted in Figures 59 and 60. For these plots, the harmonics from 1 to 10p are included in both the experimental and the predicted data. The general trends in the data are captured in the lower harmonics, but the peak magnitudes are lower than for the experimental data. This is most likely because the reverse flow region and the tip region are almost completely smoothed out, the consequence of which results in the propagation of this reduced load throughout the rotor plane. If these peaks are modeled, than the trimmer will have to increase the angle of attack in order to reach the same trim target. Since it was explained earlier that the effect of the trimmer is to increase the magnitude of the lower harmonic results, this improvement will increase the magnitude of these curves and better match the flight test data. The corresponding polar plots in Figure 61 have similar results, the reverse flow region is smoothed out, as are many of the higher fluctuations on the advancing side. However, qualitatively, the general character of the normal force results is

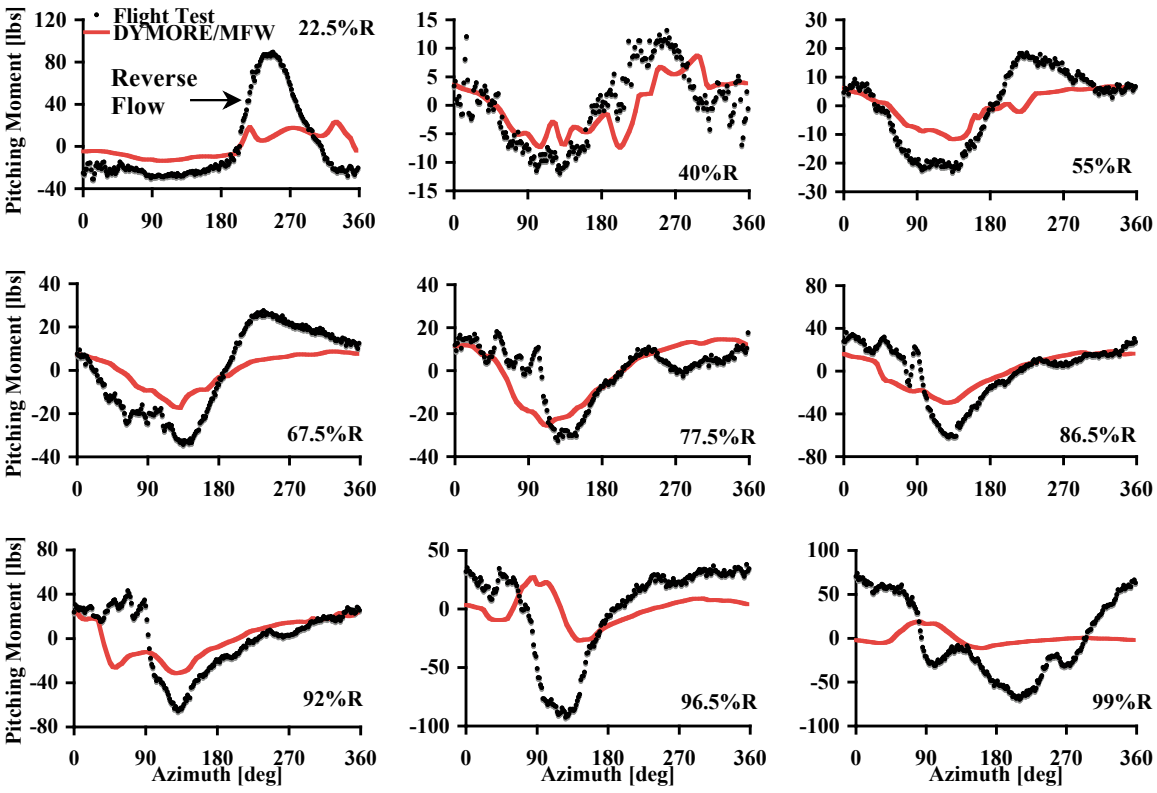


Figure 58: UH60A case 8534 mean removed pitching moment [lbs] using free wake angle of attack coupling

well captured, although the positions of the peak regions are slightly out of phase with the experimental data.

The aerodynamic results are only one aspect of the analysis; internal forces and moments of the blade can also be analyzed. Figures 62, 63, and 64 plot the internal flap, chord, and torsional moments for selected radial stations along the blade. The relative prediction of the internal moments with the experimental data is much better than for the aerodynamic loads. This is because, in part, of the use of a good structural code. However, the deficiencies in the aerodynamic loading do affect these results, and the lower magnitude of the external loading leads to lower magnitudes in the internal loading required to balance these external forces. This result appears in the chord bending moment, which has the lowest comparative magnitude to the experimental data. The loss of accuracy in

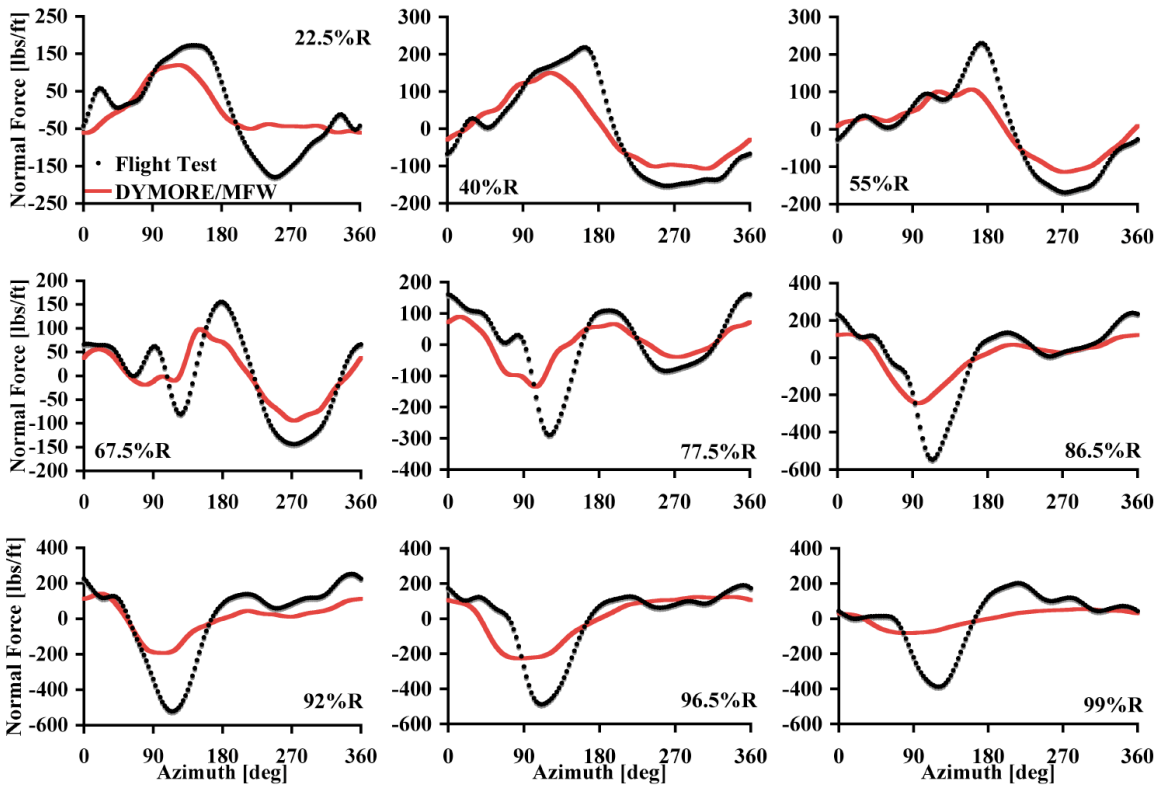


Figure 59: UH60A case 8534 1-10p normal force [lbs/ft] using free wake angle of attack coupling

the unsteady loads in the advancing region appears in the torsional moment results. The torsional moments are impacted the most by the pitching moment computations, which are the hardest to model. The inability to model the high angles of attack in the reverse flow region can be seen at about $\psi = 270^o$ for the r=22.5%R radial station.

When comparing the wake geometries of the fully-trimmed MFW and fully-trimmed DYMORE-MFW cases, the addition of elastic motion yields more physical results than the free wake code run independently. Figure 65b illustrates the resulting free wake geometry when using fully-coupled DYMORE-MFW. Although the top view resembles prescribed wake geometry, a rear-view provides a better perspective of the roll-up of the wake on either side of the rotor disk. For comparison, the fully rigid results are given in Figure 65a. These plots show that, like the elastic coupling, the top view resembles a prescribed wake, but

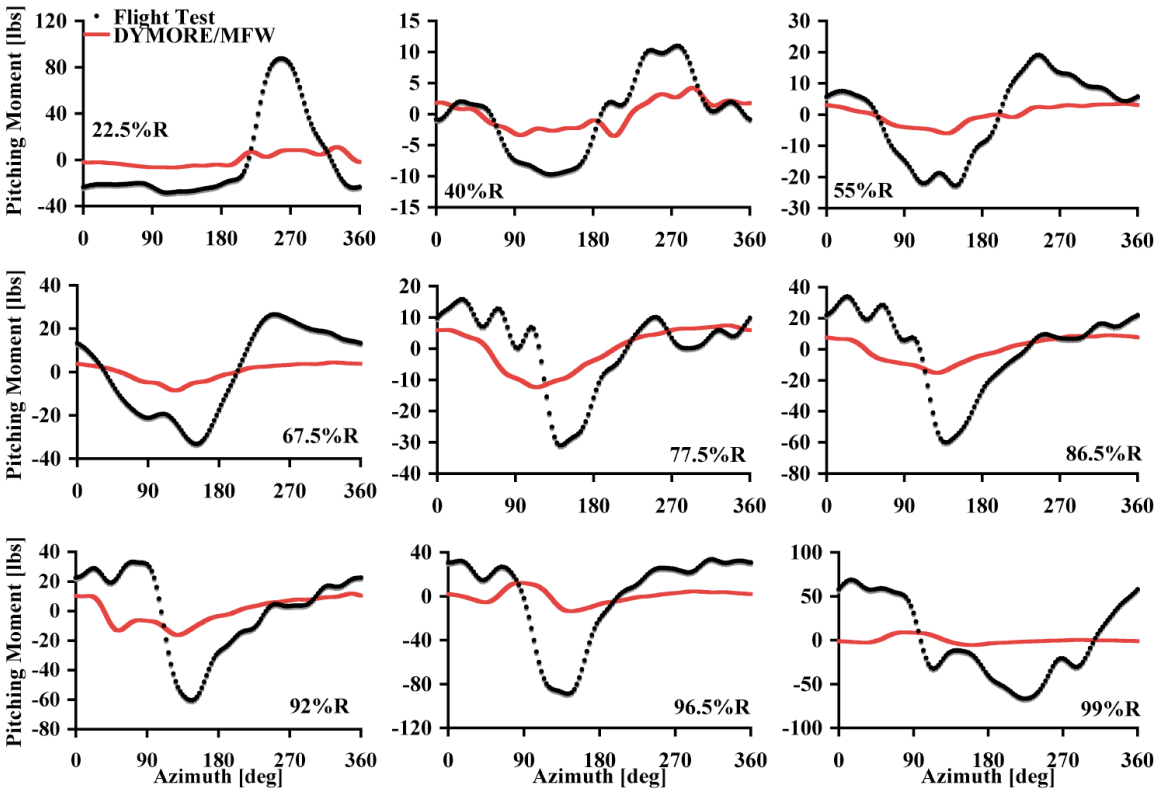


Figure 60: UH60A case 8534 1-10p pitching moment [lbs] using free wake angle of attack coupling

the rear view shows that the geometry is more complex and does exhibit roll-up. However, the fully trimmed rigid case is a non-physical result, where the left side rolls up and the right side rolls down. The elastic coupling alleviates this problem and the wake geometry is closer to the physical system.

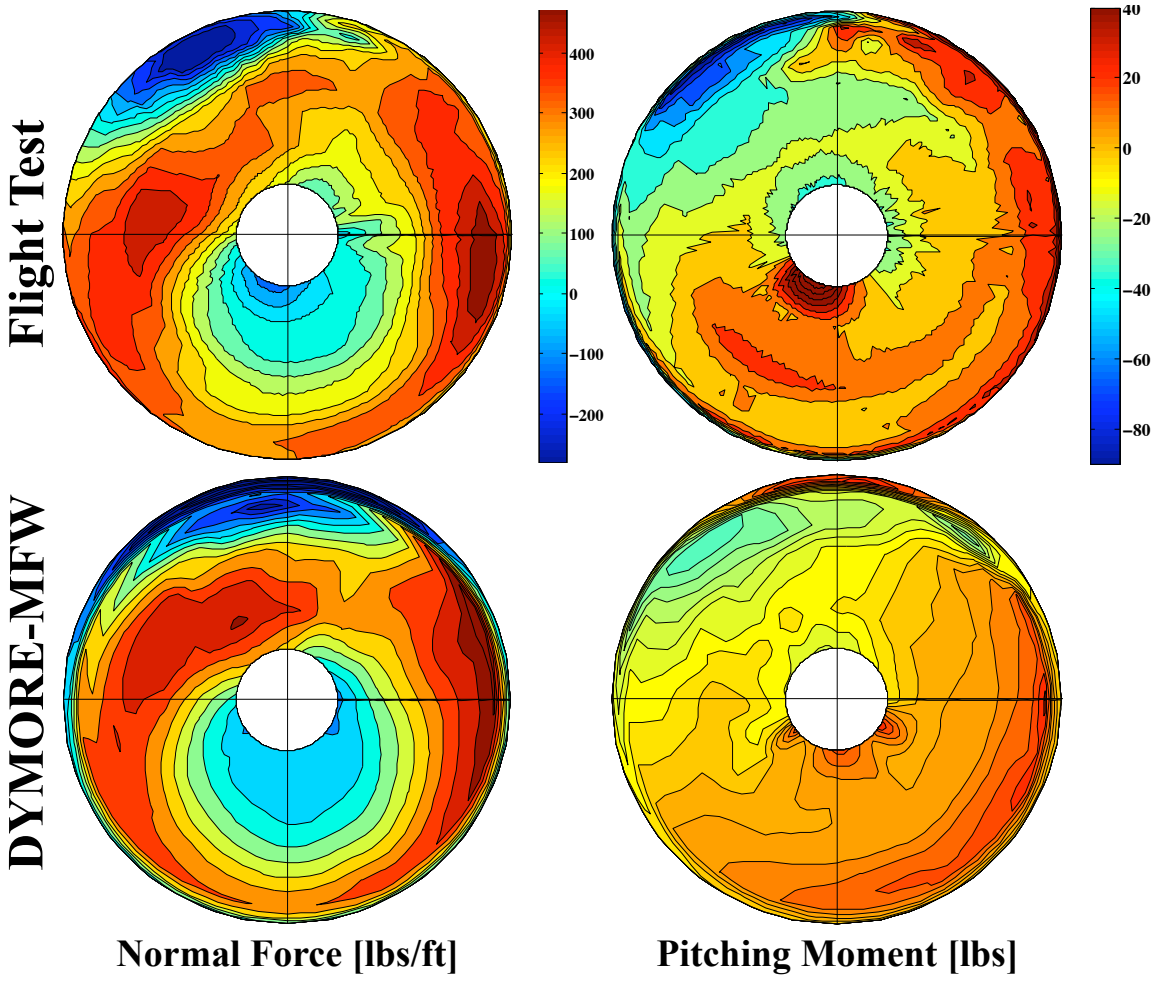


Figure 61: UH60A case 8534 normal force [lbs/ft] and mean removed pitching moment [lbs] using free wake angle of attack coupling compared with flight test data, freestream flow from left to right

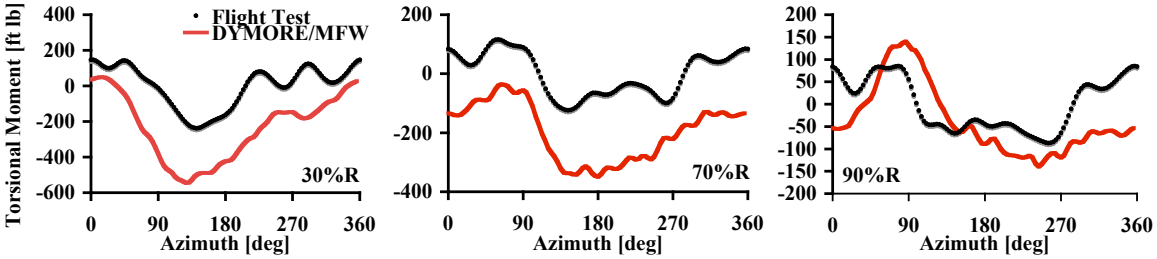


Figure 62: UH60A case 8534 torsional moment [ft lbs] using free wake angle of attack coupling

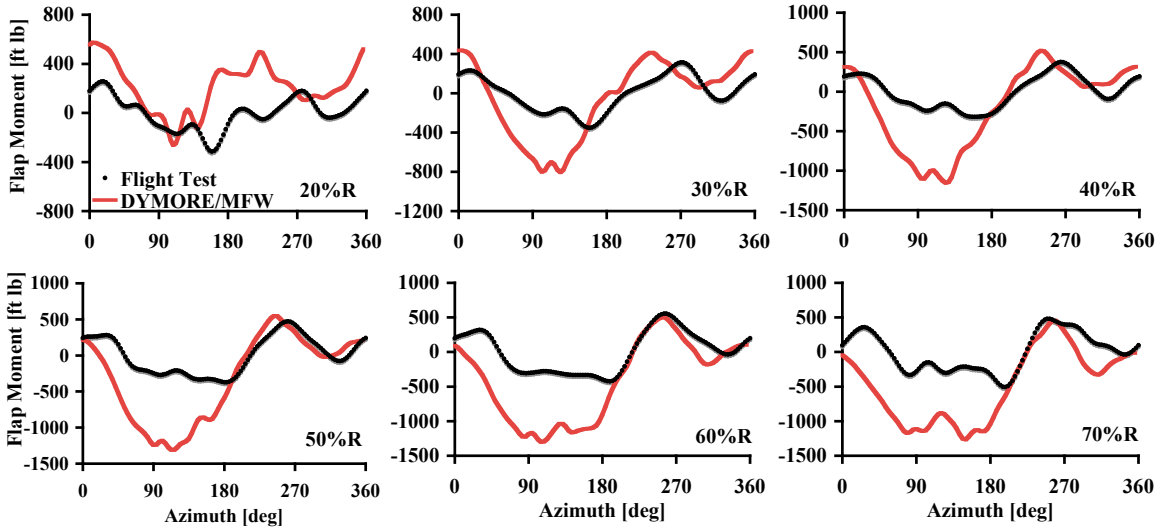


Figure 63: UH60A case 8534 flap bending moment [ft lbs] using free wake angle of attack coupling

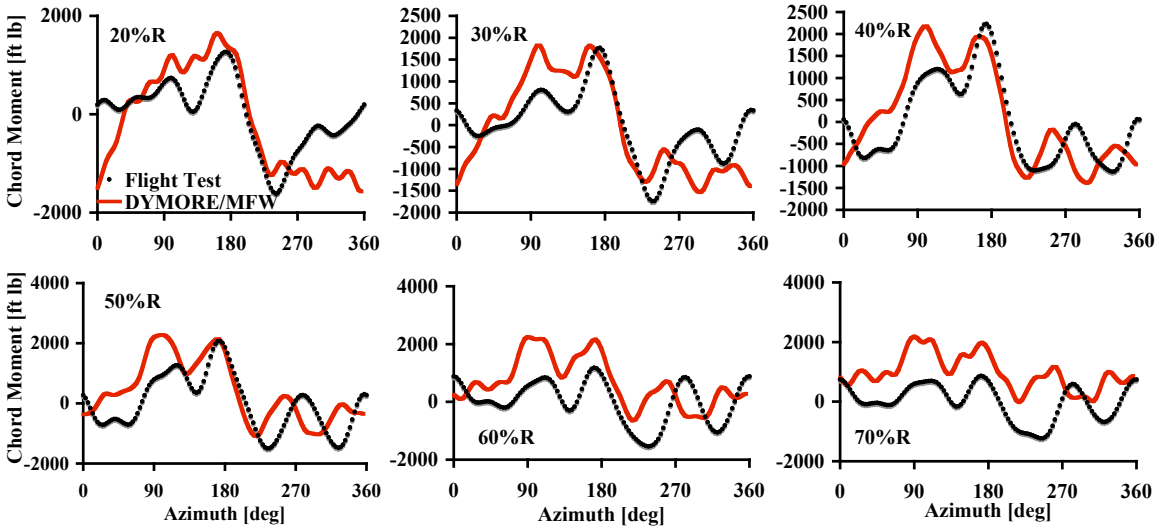
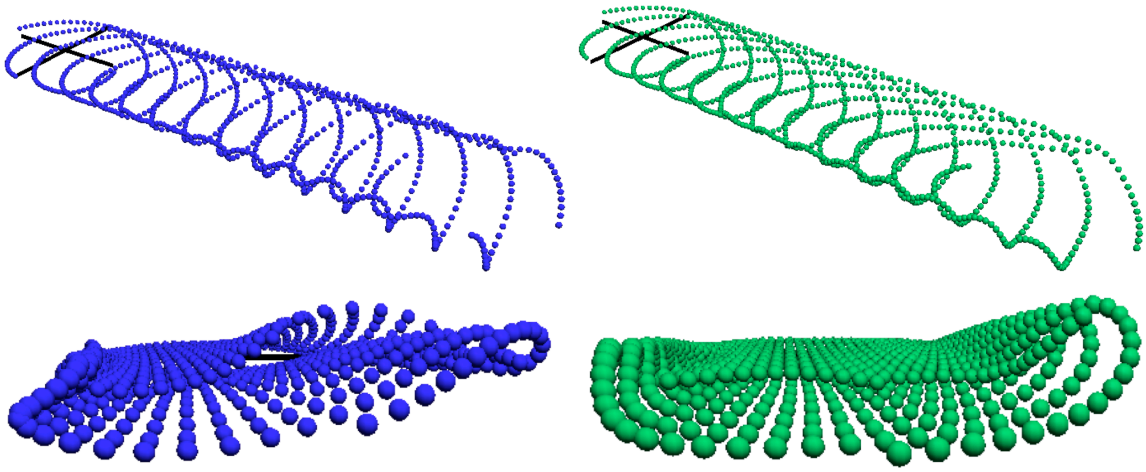


Figure 64: UH60A case 8534 chord bending moment [ft lbs] using free wake angle of attack coupling



(a) Uncoupled **(b) Coupled**

Figure 65: UH60A case 8534 comparison of wake geometries using either trimmed MFW with no external computations, or fully-coupled DYMORE-MFW

5.4 UH60A Case 9017

For this low speed, high thrust case the physics of interest are primarily the dynamic stall that occurs between about $\psi = 270^o$ to $\psi = 360^o$. This case was run using a one wake trailer free wake grid with the Peters aerodynamic model and angle of attack FSI coupling. The results in Figures 68 to 69 illustrate that dynamic stall is not being captured. Since there is no method to capture dynamic stall implemented in the simplified aerodynamic models contained in DYMORE, this result is expected. For the normal forces in Figures 66 and 68a the initial results are promising even without the dynamic stall model. The main character of the solution on the advancing side of the rotor is captured, the magnitudes and phases are off, but the peaks are present. The high loads on the retreating side of the blade near the tip region are present as are the loads in the low region near $\psi = 180^o$, although these peaks are also out of phase with the experimental data. The lower magnitude forces, indicated at the center of the experimental polar plot, are not modeled. However, the line plots show that this is because of the mean loading and not because the main features are being missed. The peak normal force loading is concentrated in the region around $\psi = 0^o$ and $\psi = 180^o$. The phase is different, but the character is the same. The dynamic stall region seen in the 4^{th} quadrant is missed by the computations. The pitching moment plots in Figures 67 and 68b also have some of the same characteristics. The high moments are concentrated in the 1^{st} quadrant and the lowest moments are in the 3^{rd} quadrant, for both the computations and the flight test.

The wake geometry also provides interesting results. The propagation of the vortices into the far field is slower than for the 8534 case. Since this case has a lower advance ratio this is a correct result. The rear view illustrates a roll-up region on the advancing side of the blade. Being a high thrust case, 9017 has higher loads on the rotor blades and thus a higher pressure differential on the upper and lower blade surfaces. This translates into stronger tip vortices, which results in stronger roll-up of the wake where the loads are the highest.

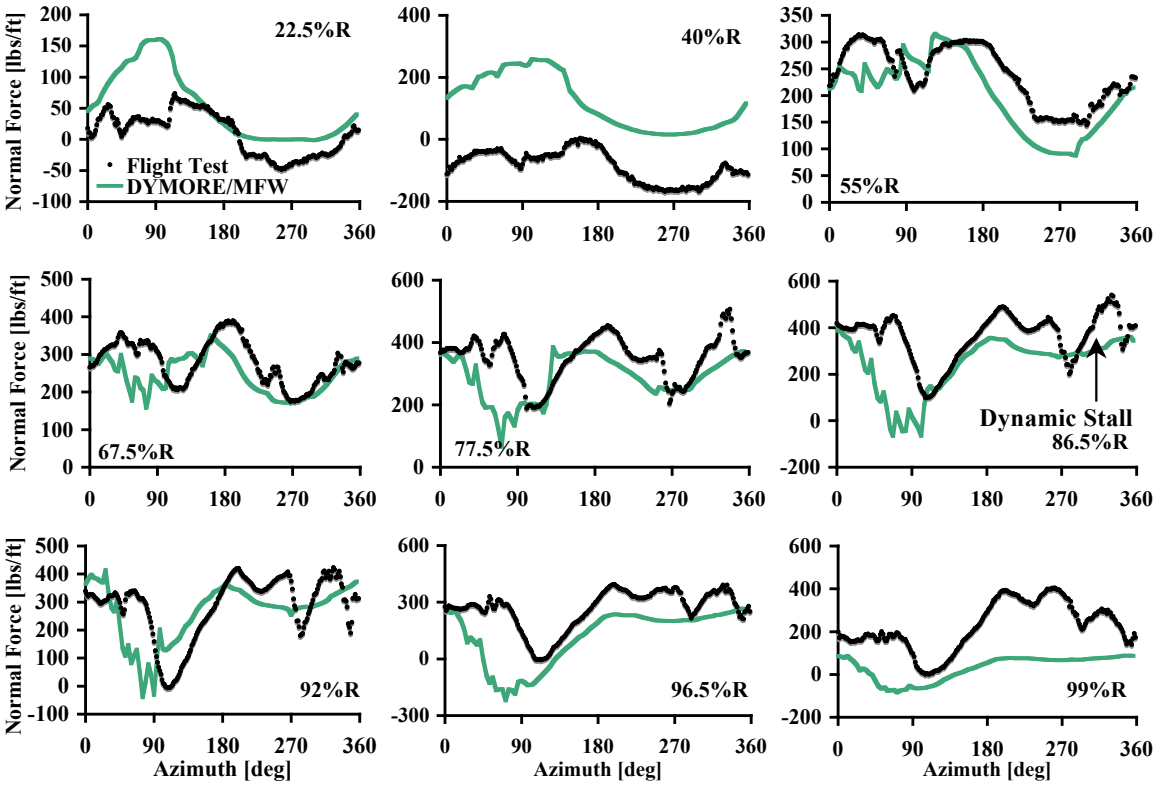


Figure 66: UH60A case 9017 normal force [lbs/ft] using free wake angle of attack coupling

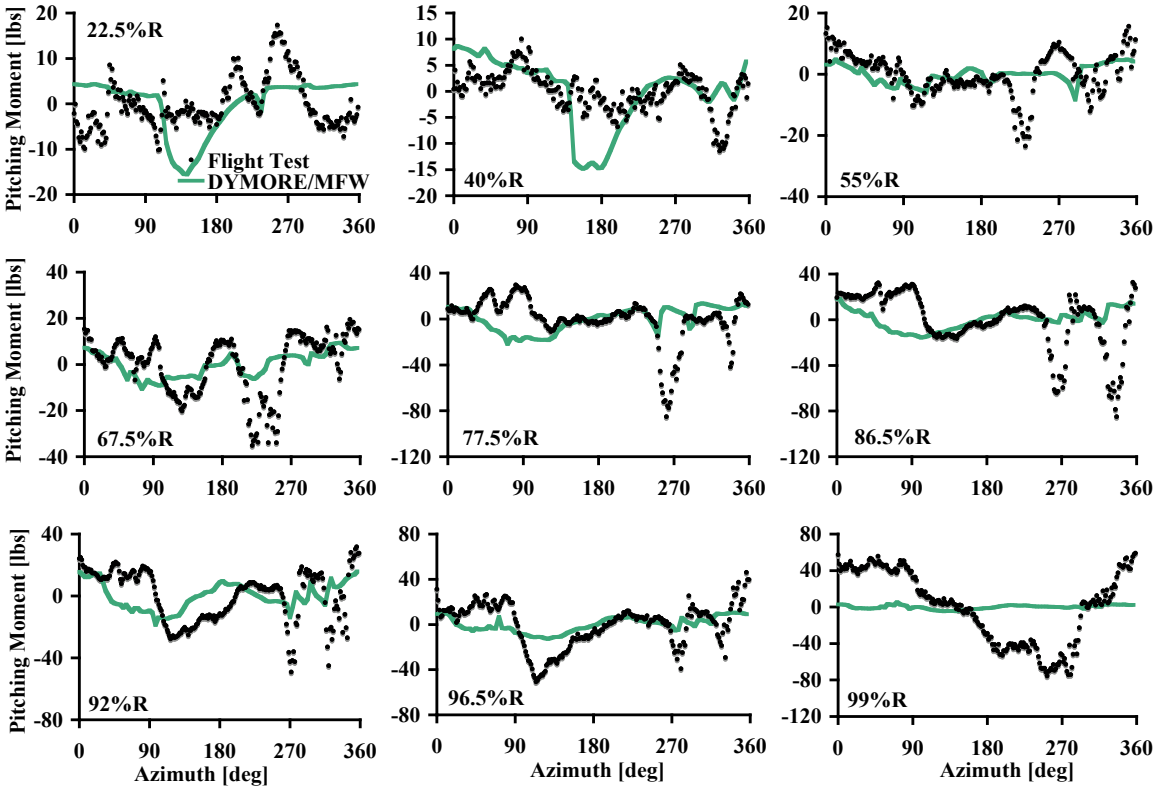


Figure 67: UH60A case 9017 mean removed pitching moment [lbs] using free wake angle of attack coupling

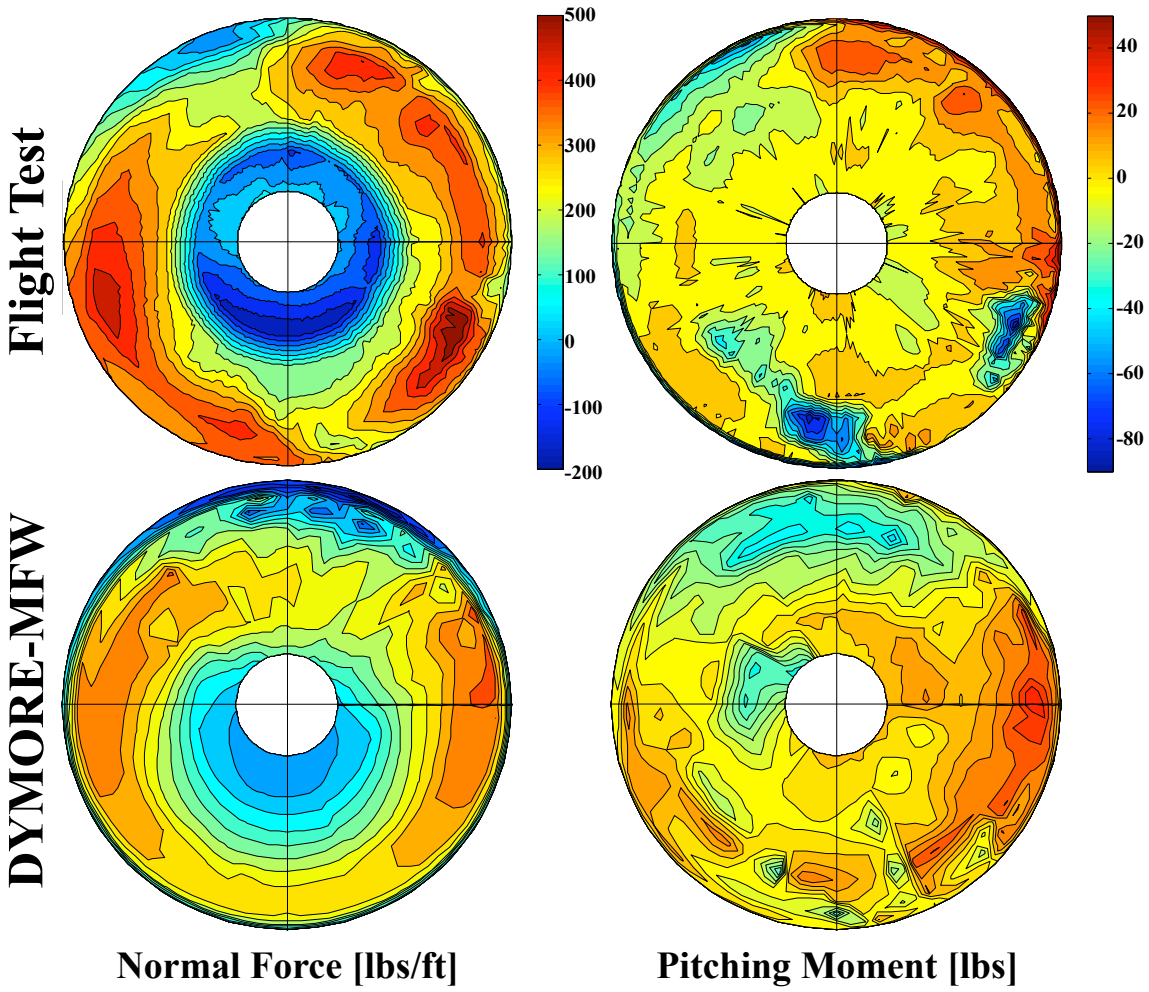


Figure 68: UH60A case 9017 normal force [lbs/ft] and mean removed pitching moment [lbs] using free wake angle of attack coupling compared with flight test data, freestream flow from left to right

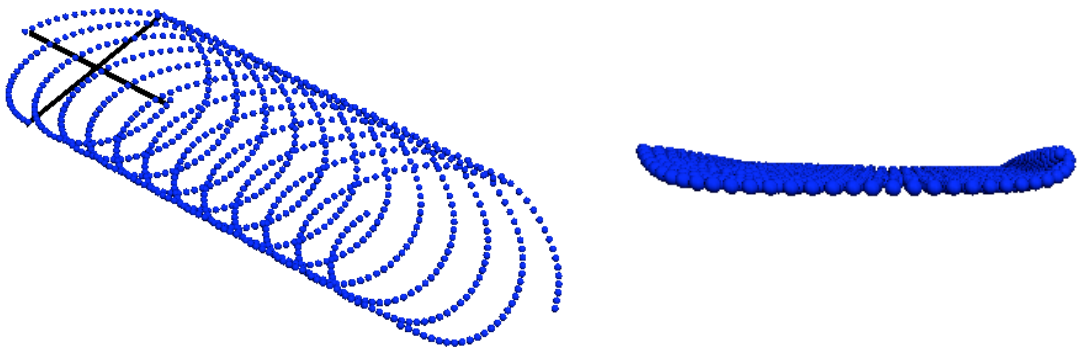


Figure 69: UH60A case 9017 free wake geometry

5.5 UH60A Case 8424

Case 8424 is a forward flight case closer to the center of the flight envelope, characterized by mild vibratory loading on the advancing blade, and reverse flow on the retreating blade. The flight test data plotted in Figure 70 in polar plot form and the same data plotted in Figures 71 and 72 in line plot form illustrate these characteristics. The vibratory loading is located by the higher frequency fluctuations near 67.5% radius location. The reverse flow region is characterized by the negative loading near the root of the blade around an azimuthal position of 270^o. The results presented in this section are generated using a DYMORE-MFW angle of attack coupling scheme using a Peters 2D aerodynamic model. The simulation is run to trim convergence using the targets given in Table 3.

The results for the aerodynamic loading are presented in two forms in Figures 70, 71, and 72. These results show many similar characteristics to the results of case 8534 presented in section 5.3. The prediction of the normal forces does not match the vibratory predictions on the advancing blade, instead the predictions show lower magnitude higher frequency fluctuations in this region. The simplified 2D aerodynamic model used to model this case is insufficient to capture the vibratory loading on the blade. The reverse flow region, which was not previously captured in the 8534 case, has been captured in the 8424 case. The slower forward speed of this flight condition reduces the size and severity of this region making it easier to model the forces in this region. Overall, there is a phase lag in the predictions on the advancing blade. The pitching moment computations, which are generally more difficult to predict because of the added variable of the moment arm, is not captured at all using the 2D aerodynamics.

The prediction of the internal structural moments is influenced by the external aerodynamic loading applied to the blades. The flap bending, chord bending, and torsional moments are plotted in Figures 73, 74, and 75 respectively. The flap bending moment prediction has the same general character as the flight test data. however, there is a phase lag in this data which matches the phase lag in the external aerodynamic loading. The chord

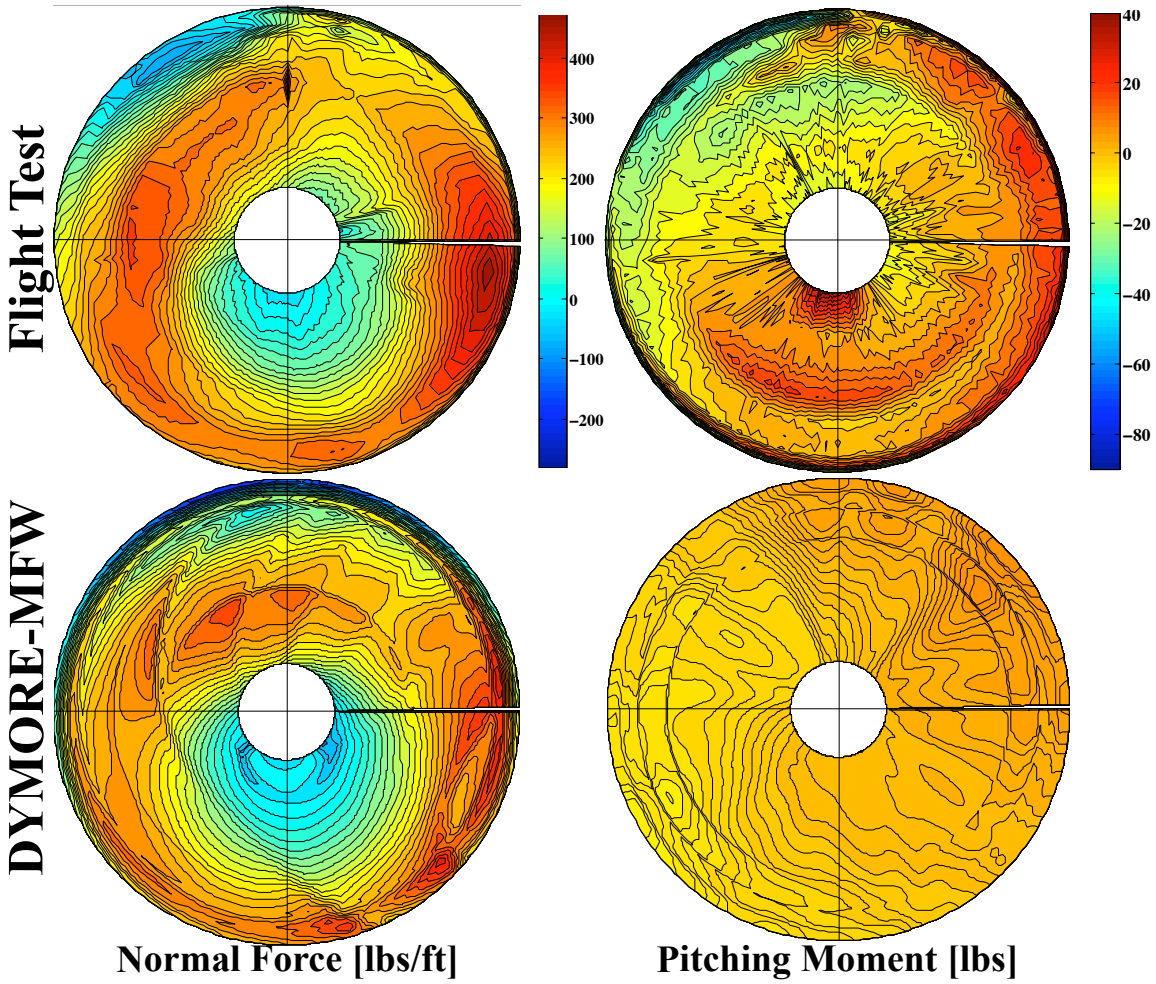


Figure 70: UH60A case 8424 normal force [lbs/ft] and mean removed pitching moment [lbs] using free wake angle of attack coupling compared with flight test data, freestream flow from left to right

moment has a lower magnitude than the flight test data and much more higher harmonic content. This moment is in reaction to the unsteady motion of the blade sections relative to the blade axis. This moment has the highest vibrations near the ends of the blade and is more steady near the center of the blade. The torsional moment, in reaction to the blade twisting motion, matches the flight test data motion, but has a phase lag over the flight test data.

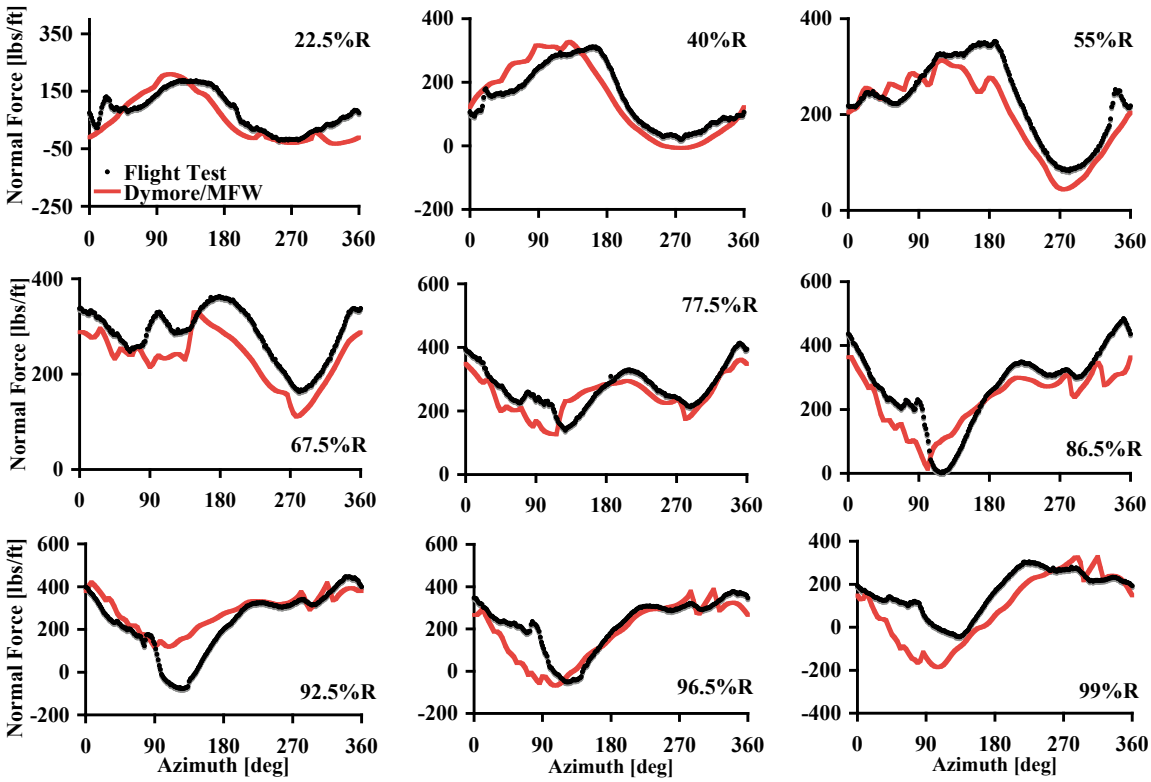


Figure 71: UH60A case 8424 normal force [lbs/ft] using free wake angle of attack coupling

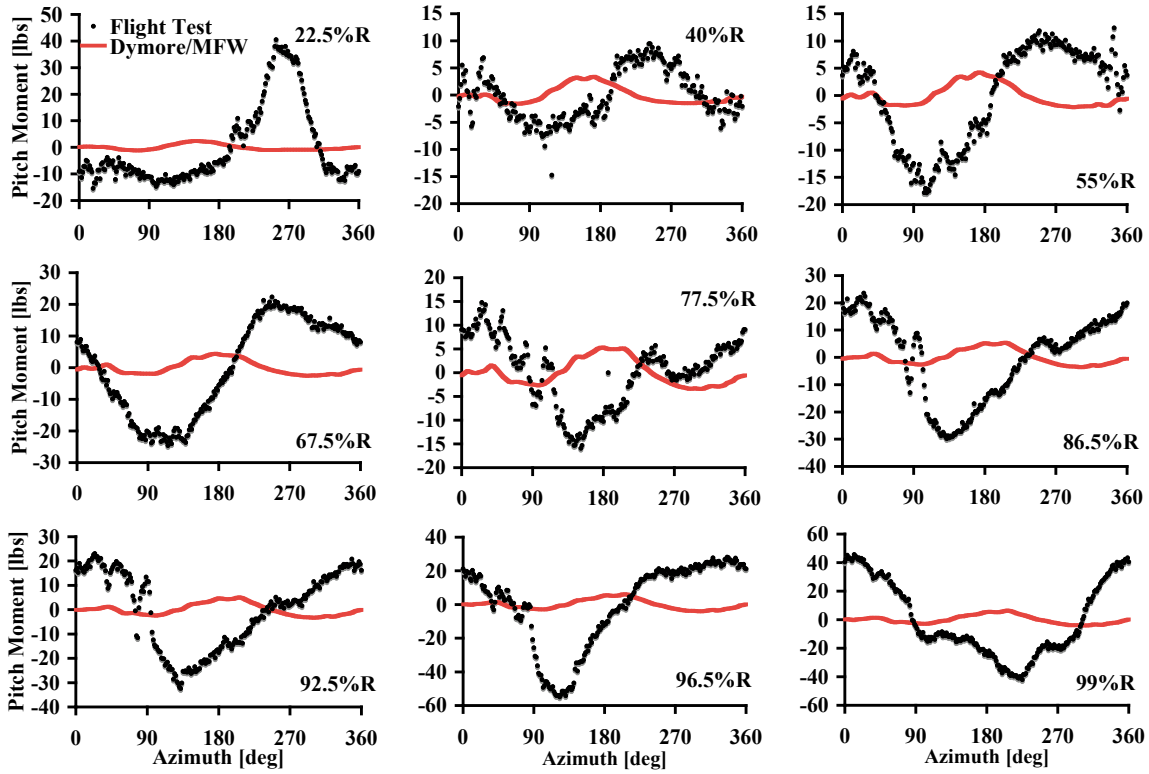


Figure 72: UH60A case 8424 mean removed pitching moment [lbs] using free wake angle of attack coupling

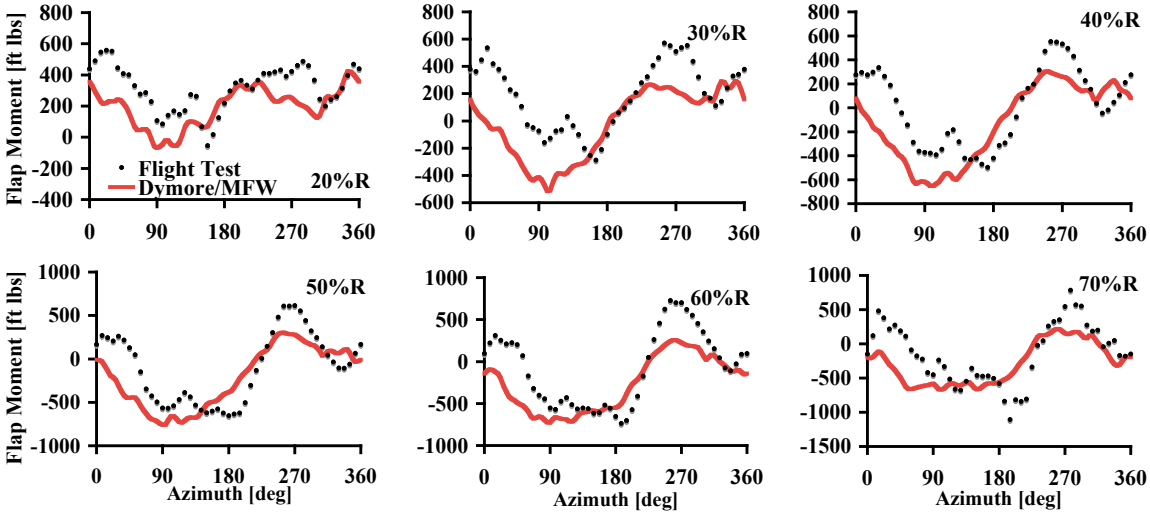


Figure 73: UH60A case 8424 flap bending moment [ft lbs] using free wake angle of attack coupling

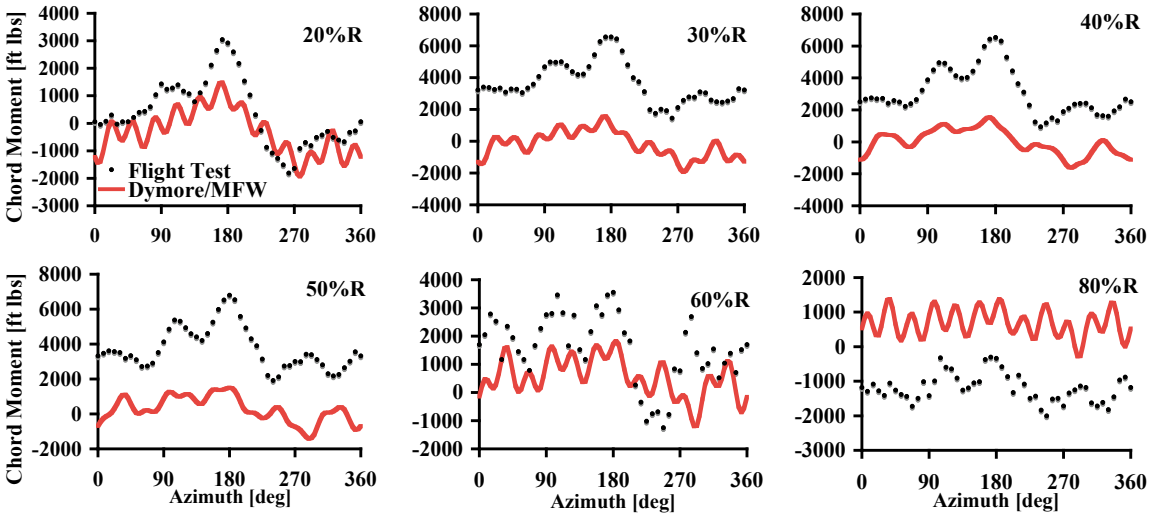


Figure 74: UH60A case 8424 chord bending moment [ft lbs] using free wake angle of attack coupling

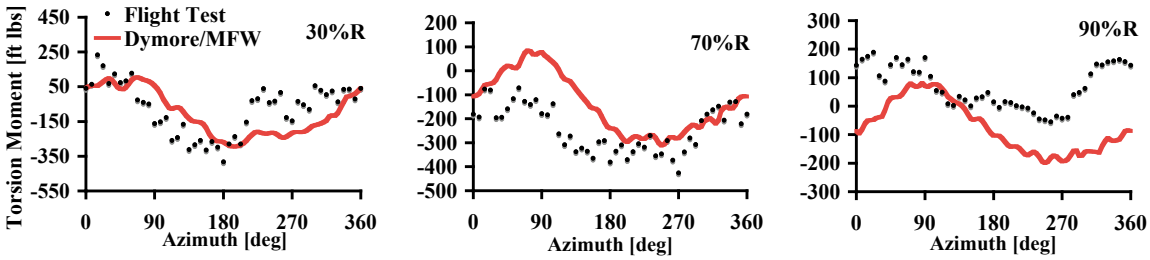


Figure 75: UH60A case 8424 torsional moment [ft lbs] using free wake angle of attack coupling

CHAPTER 6

RESULTS: ROTOR-FUSELAGE CFD STUDIES

6.1 GT Unsteady Low Mach Number Preconditioning

The application of low Mach number preconditioning is useful for those cases where the local Mach number is below the compressible boundary of around $M = 0.3$. The GT case has been selected for the analysis of the benefits of using low Mach number preconditioning because both the freestream velocity, and the blade tip speed for this case are below the compressible limit $\left(M_\infty = 0.0295, M_{tip} = 0.295\right)$. The consequences of using the fully-compressible URANS equations on this case can be seen in the surface Cp slices in Figure 76. The red curves indicate the un-preconditioned results. These curves have clear non-physical numerical fluctuations that almost completely disappear when compared to the blue curves, which represent the preconditioned results. The smoother results better represent the physical situation where the flow features change continuously along the surface. A better comparison of the impact of using the preconditioning is illustrated in Figure 77. These images of the surface Cp at different time steps provide a more complete description of what is happening. The preconditioned method shows improvement in both the physical results and the numerical quality. The numerical fluctuations seen in the un-preconditioned data in the nose region of $\psi = 995^o$ and $\psi = 1035^o$ are reduced when compared to the preconditioned results. The vortex impact seen at about $x = 0.2R$ is much stronger in the preconditioned results as is the vortex located at about $x = 3R$. The vortex impact at $x = 3R$ is numerically more significant, because physically the vortex has been propagating longer and has had more time to dissipate both physically and numerically. So the ability to maintain an ”older” vortex segment in the prediction is more significant than a recently shed one.

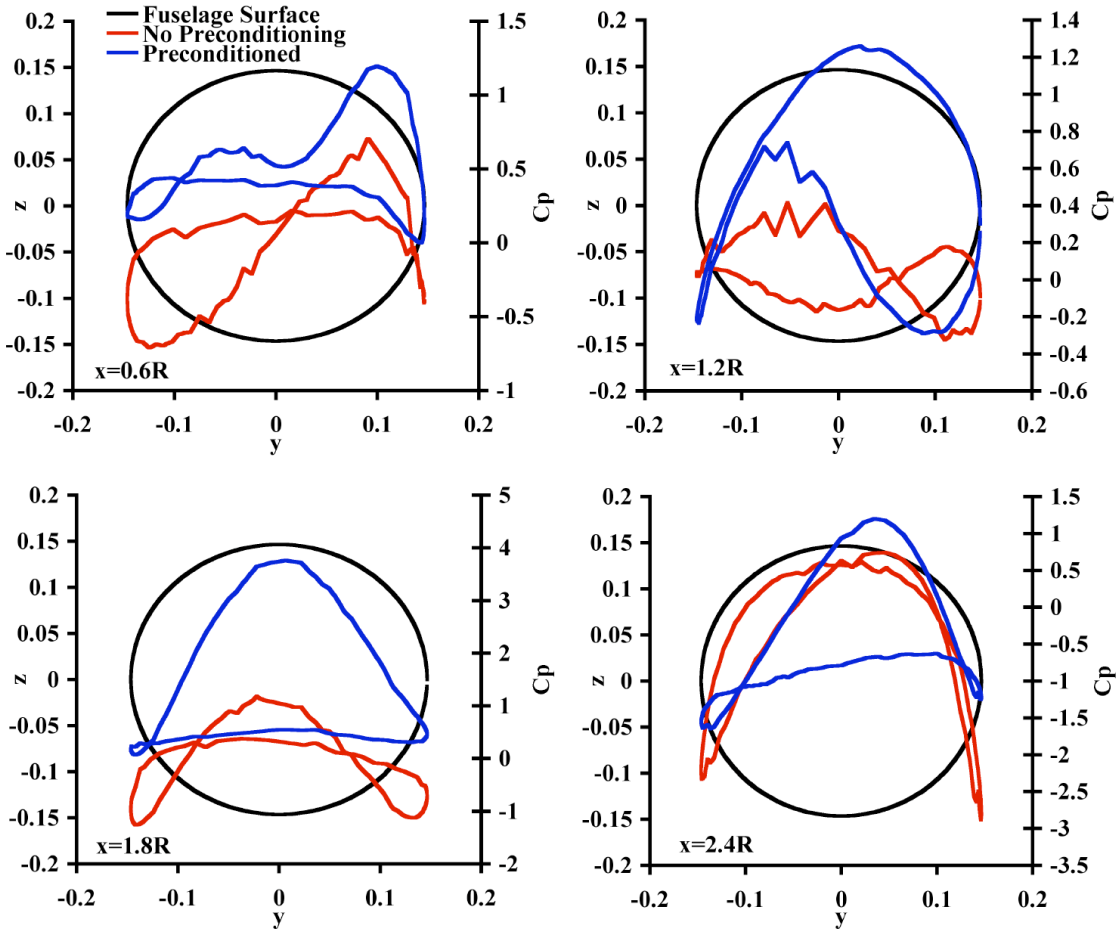
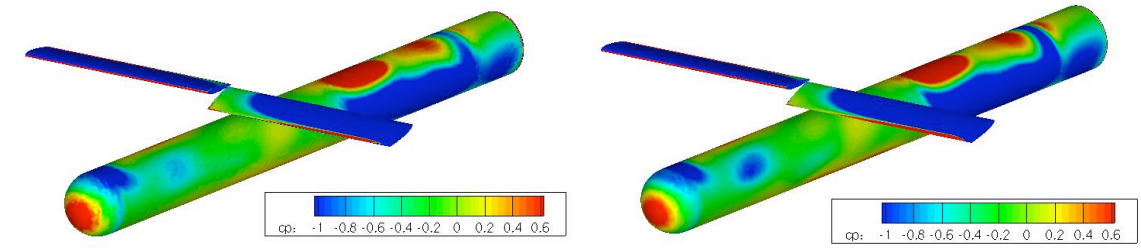
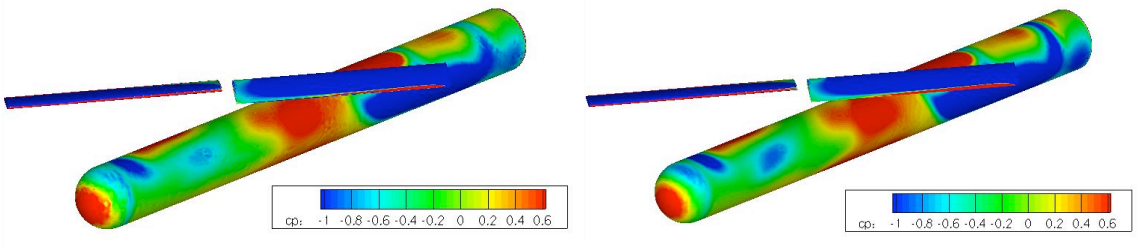


Figure 76: GT slices of surface Cp at different x locations along the fuselage for $\psi = 1080^o$ comparing the compressible RANS results with the low Mach number preconditioned RANS method



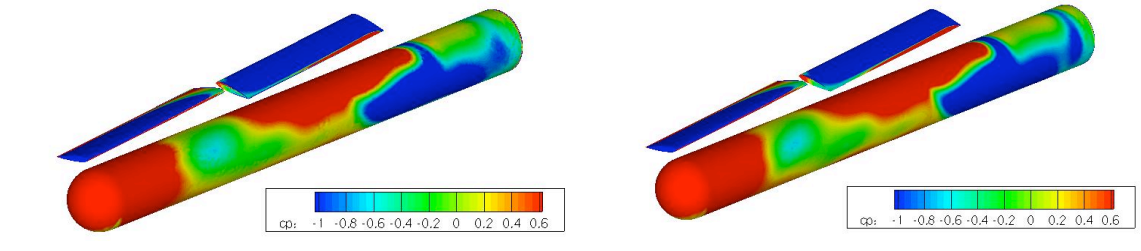
No Preconditioning **Preconditioned**

(a) $\psi = 995^o$



No Preconditioning **Preconditioned**

(b) $\psi = 1035^o$



No Preconditioning **Preconditioned**

(c) $\psi = 1080^o$

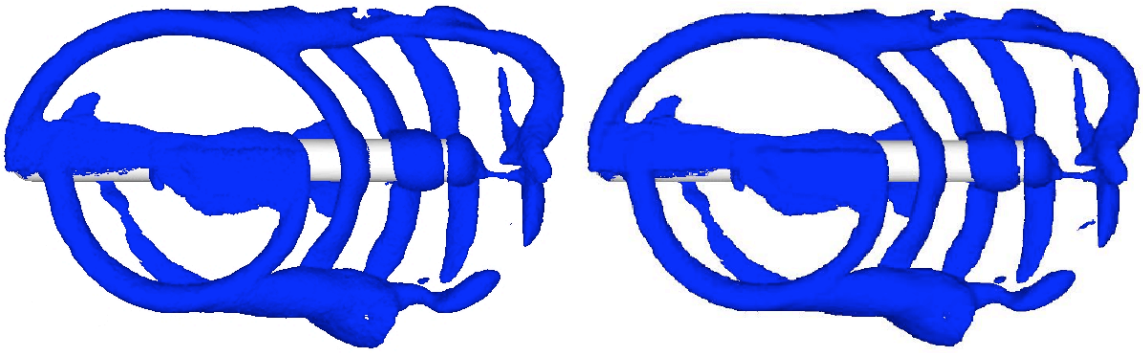
Figure 77: GT surface Cp distributions comparing both the un-preconditioned and the preconditioned cases

6.2 GT Rigid Motion

The use of a rigid motion algorithm is a first step toward increasing the accuracy of the structural computations in the CFD code. This method enables prescribed rigid motions to be used to control the rotor blade motion. It is a step closer to real blade motion than using an un-articulated model, and a step further from real blade motion than an elastic blade model. This method is implemented entirely within the CFD code, and thus data need not be passed from a structural code. This rigid motion algorithm will be used to model three different systems of varying levels of articulation. The first, least-articulated, case will be the GT rotor-fuselage model. This is a teetering rotor that exhibits only flapping motion. The second will be the ROBIN rotor-fuselage model. This case has flap, pitch, and lead-lag components. However, the lead-lag component is a small constant contribution. The last case included in the next chapter is the fully-articulated UH60A rotor case. This case has all levels of articulation with higher harmonic content, and is the most comprehensive case presented.

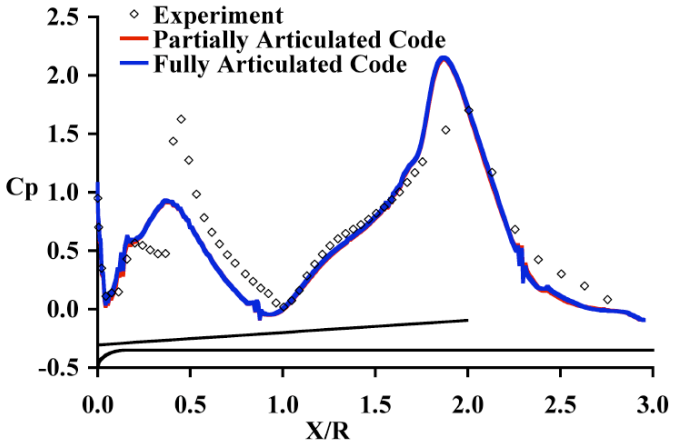
The GT case has the simplest motion of all cases presented and thus is the easiest case to model. This case was run for a time step size of 1^o over three revolutions using the compressible Euler equations with 2^{nd} order time-accuracy with local time-stepping and 10 subiterations. The reason for choosing this configuration of the governing equations is to directly compare this case with a similar case that was run with a partially-articulated motion algorithm [2]. The partially-articulated algorithm has the ability to model all articulation necessary for this case; thus, this system becomes an ideal validation case. A comparison of these two algorithms with the experimental data is plotted in Figure 79. This figure plots the 3^{rd} revolution time-averaged Cp distributions along lengthwise cuts of the fuselage. In a comparison between the fully-articulated and partially-articulated algorithms, identical results are obtained. The results are also identical for the iso-vorticity plots in Figure 78. This proves that the implementation of the new motion algorithm computes the same motion as the partially-articulated algorithm for this case. The comparison with the experimental data

shows that this case is capturing all main features of the time-averaged surface pressures, resulting from the impact of the rotor wake vortices with the fuselage surface over one revolution. Figure 78 provides a snapshot of the wake at $\psi = 1080^o$. For three revolutions the wake will only extend back by four vortex loops since each of the rotor blades has passed the same point only two times. All four of these loops are present in the iso-vorticity plots.

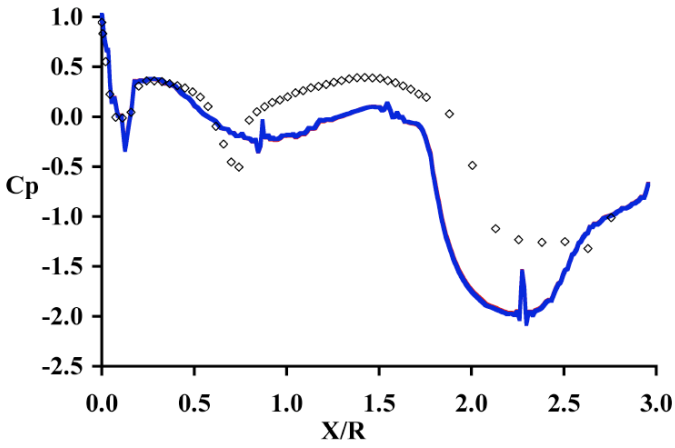


(a) Partial Articulation **(b) Full Articulation**

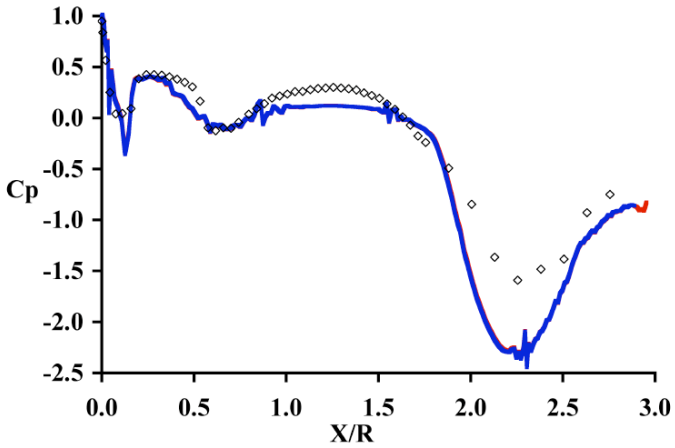
Figure 78: GT iso-vorticity plots of magnitude 0.25 at $\psi = 1080^\circ$ comparing the partially-articulated code and the fully-articulated code



(a) Top Centerline



(b) Left Centerline



(c) Right Centerline

Figure 79: GT fuselage time averaged fuselage centerline Cp distributions

6.3 ROBIN Rigid Motion

The ROBIN system is another rotor-fuselage interaction case. However, this system includes more articulation than the GT case just shown. This model was run using the 2^{nd} order time-accurate compressible URANS equations with the Spalart-Allmaras turbulence model. The system was run with a 1^o time step size and uses local time stepping with 10 subiterations. The experimental data used for the comparison comes from the Mineck and Althoff experiment [85]. In this document, the standard pressure coefficient is not plotted to avoid an undefined result when the freestream dynamic pressure becomes zero during hover. Instead, a modified pressure coefficient is used, which is the standard pressure coefficient multiplied by a factor of $\mu^2/2$. The difference between the corrected and the published data plotted in Figures 82 and 83 is a phase shift of 28^o. This phase shift is included to account for the difference in the measurement location versus the measured blade position. This correction is accounted for in reference [2].

A comparison with the corrected experimental data illustrates that the phase is close to the experimental data, but while the amplitude of the results is close, the mean is consistently higher than the experimental data. This over-prediction of the steady-state pressure is attributed to using a blade pitch setting that is too high. The reason for this high blade pitch is attributed to either an aspect of the system that is not being modeled correctly, or to uncertainty in the experimental blade pitch settings. Using a trim model would alleviate this problem by computing the correct blade pitch needed to obtain the desired thrust. However, the CFD code does not have a trim model implemented, so the current blade pitch settings are obtained from a simulation run by reference [87]. A comparison of modified pressure coefficient at various points on the fuselage surface has very little difference with respect to the computed results obtained in reference [2]. The differences between the two cases are that a small lead-lag angle is added to the motion, and the hinge offsets are included. These changes shift the rotor motions closer to the fuselage surface by 0.00133 grid units, and add a phase shift to the blades of 0.95^o, the remaining motions are left unchanged. These

additional parameters shift the rotor wake relative to the fuselage surface, and thus change the influence that the rotor wake has on the surface.

Figures 82 and 83 compare the modified pressure coefficient for four different locations along the fuselage upper centerline. These locations are illustrated in Figure 81, where the first two points are near the nose, and the last two are closer to the hub and tail respectively. The biggest difference in the results from adding the small lead-lag and hinge offsets is to smooth out the peak values of the Cp plots at x=0.201. This is the location under the leading blade, and the peaks of the pressure coefficient occur at 0^o, 90^o, 180^o, and 270^o, whenever a blade passes over this point on the fuselage. Since there is no direct impact of the wake with the fuselage at this location, the reduction in the peak indicates that the magnitude of the rotor downwash when a blade passes over this point of the fuselage has decreased, decreasing the pressure at this point. The time average of this pressure data along the upper fuselage centerline, plotted in Figure 80, shows almost identical results between the two cases, with small favorable shifts near the nose and tail using the fully-articulated case. The comparison of the iso-vorticity surfaces in Figure 84 shows that the results are almost the same in the flow field as well as on the surface. Thus, the added small changes to the blade angles have little effect on the results in this case, and neglecting them is an acceptable assumption to make in order to simplify the problem. This argument will not apply to all cases, especially the UH60A, since not all configurations use reduced articulation.

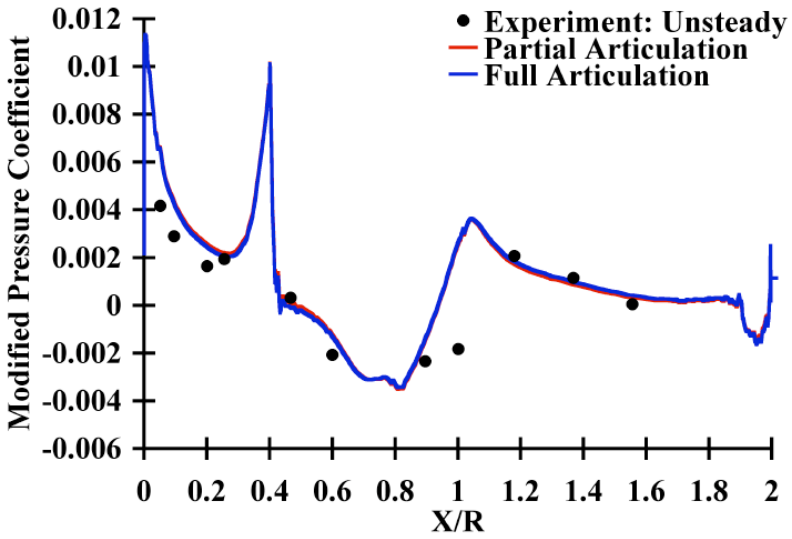


Figure 80: ROBIN time averaged centerline Cp

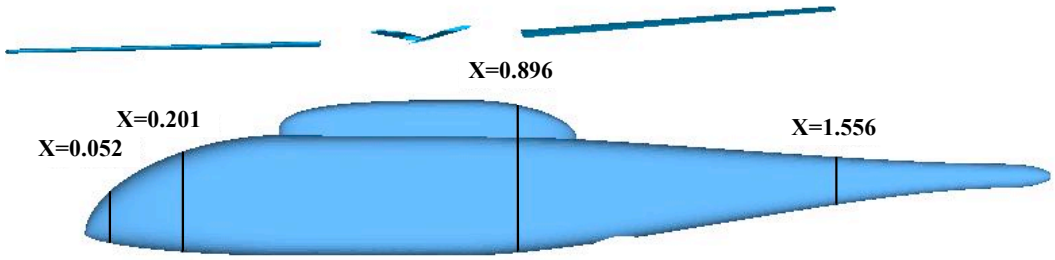
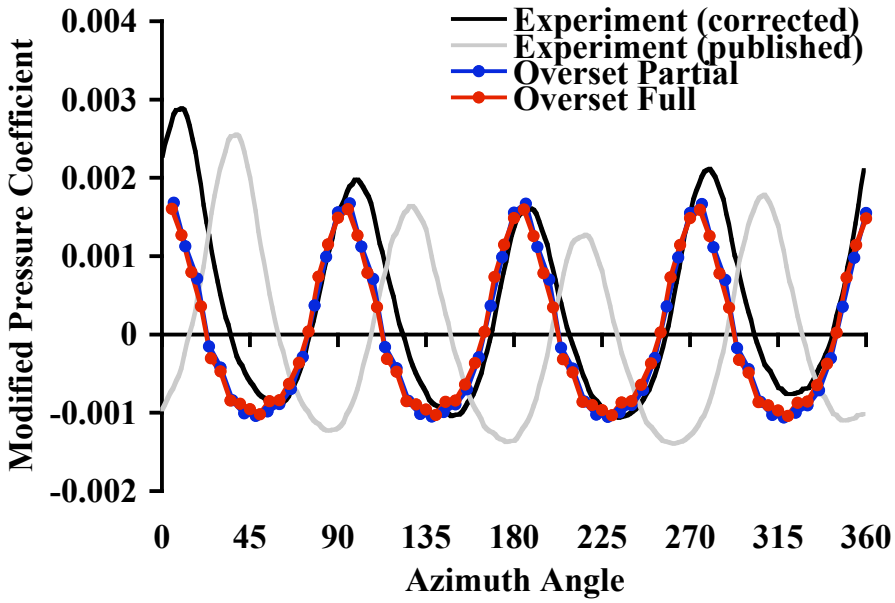
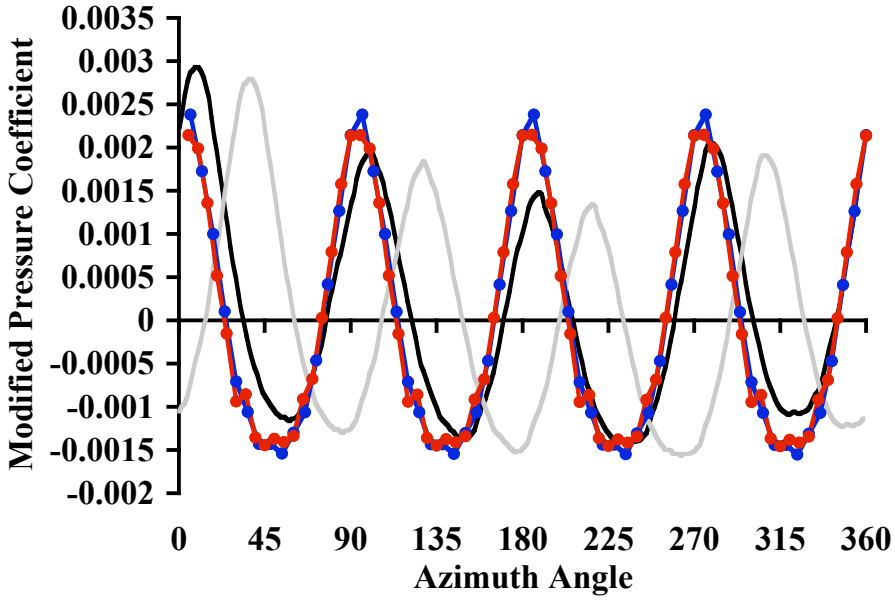


Figure 81: ROBIN fuselage pressure tap locations for data in Figures 82 and 83

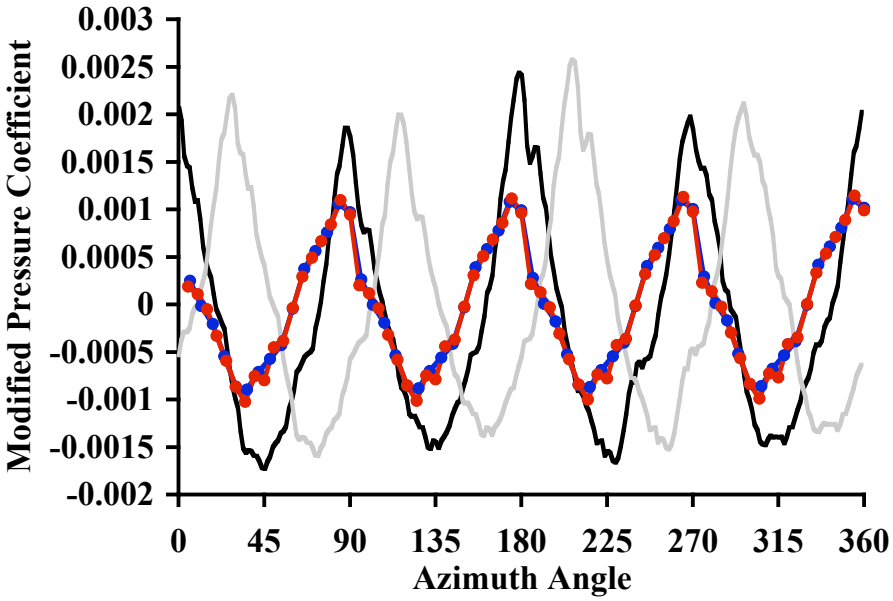


(a) X=0.052

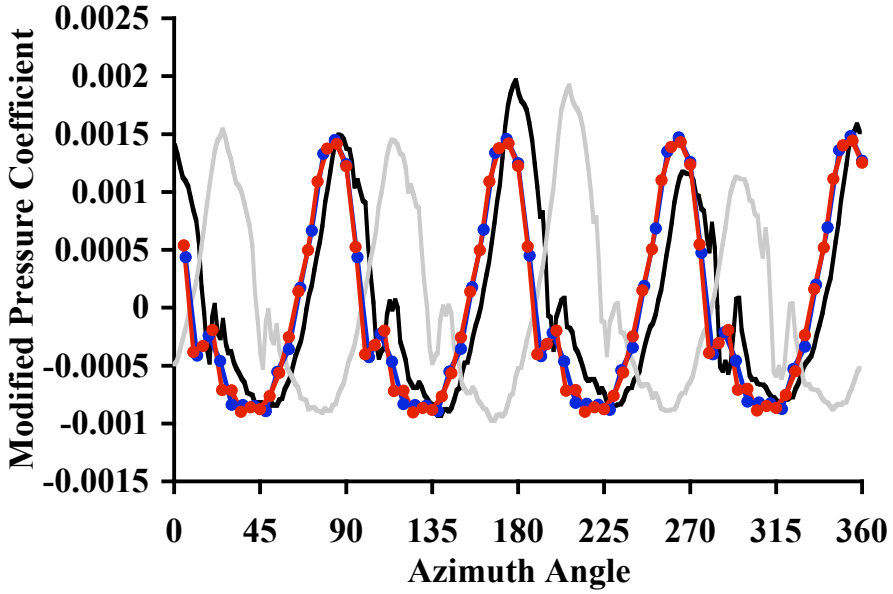


(b) X=0.201

Figure 82: ROBIN fuselage mean removed modified pressure coefficient on the fuselage centerline

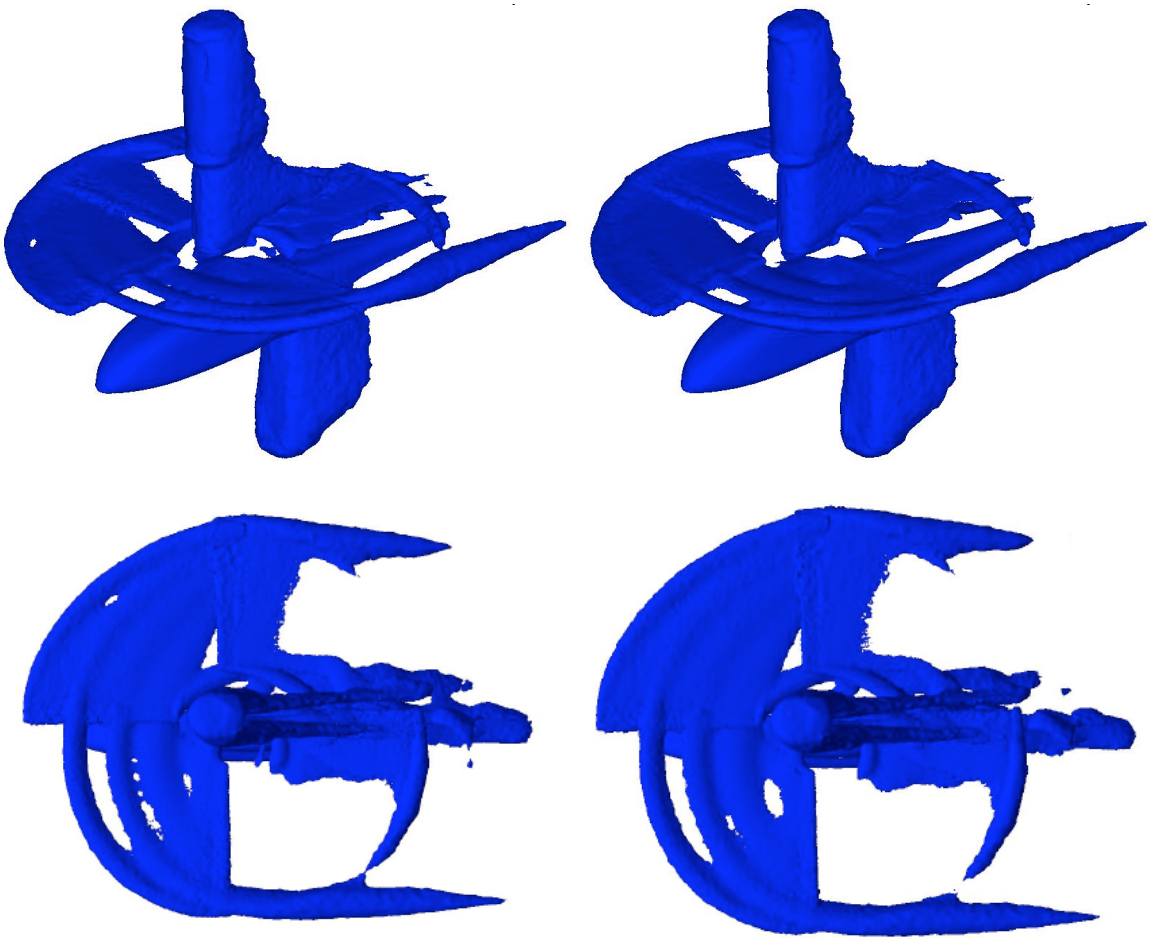


(a) X=0.896



(b) X=1.556

Figure 83: ROBIN fuselage mean removed modified pressure coefficient on the fuselage centerline



(a) Partial Articulation **(b) Full Articulation**

Figure 84: ROBIN iso-vorticity of magnitude 0.65 comparing partial and full articulation cases

CHAPTER 7

RESULTS: ISOLATED ROTOR CFD STUDIES

7.1 UH60A Rigid Motion Case 8534

As opposed to the previous cases shown, the UH60A rotor is a fully-articulated system that exhibits all possible motions, and thus requires that all the hinge motions are modeled. The articulated motion consists of pitch, flap, and lead-lag components with non-zero higher harmonic content, as well as hinge offsets and shaft tilt. The flight conditions for this case are the counter 8534 conditions in Table 2. As opposed to the ROBIN and GT cases, which are wind tunnel models, this configuration represents a full scale helicopter rotor and all experimental data used for comparison was obtained through flight test [74]. Comparison to structured CFD computations using OVERFLOW is provided. These computations were obtained from reference [3]. The unstructured and structured CFD solver settings are discussed in sections 4.1.3 and 4.1.4, the grid differences are discussed in section 4.1.5.

The elastic deformations of this full scale system are larger than for the smaller wind tunnel models studied earlier, thus applying a rigid algorithm to model the blades only approximates the actual blade motions. The impact of this approximation is illustrated in Figure 85 comparing the normal force and pitching moment results to flight test data. The results near the root of the blade, where the effects of elastic deformation are minimal, more closely match the experimental data than the results near the tip of the blade, where the elastic deformation is more significant. This impact is greatest at about $\psi = 135^o$ near the tip of the blade where the negative lift indicated by the flight test data is predicted as a large positive lift by the CFD computations. The rigid blade motion used in the CFD computations does not model the elastic twist of the blade, which decreases the angle of attack near the tip, resulting in higher lift in this region than the measured data. This

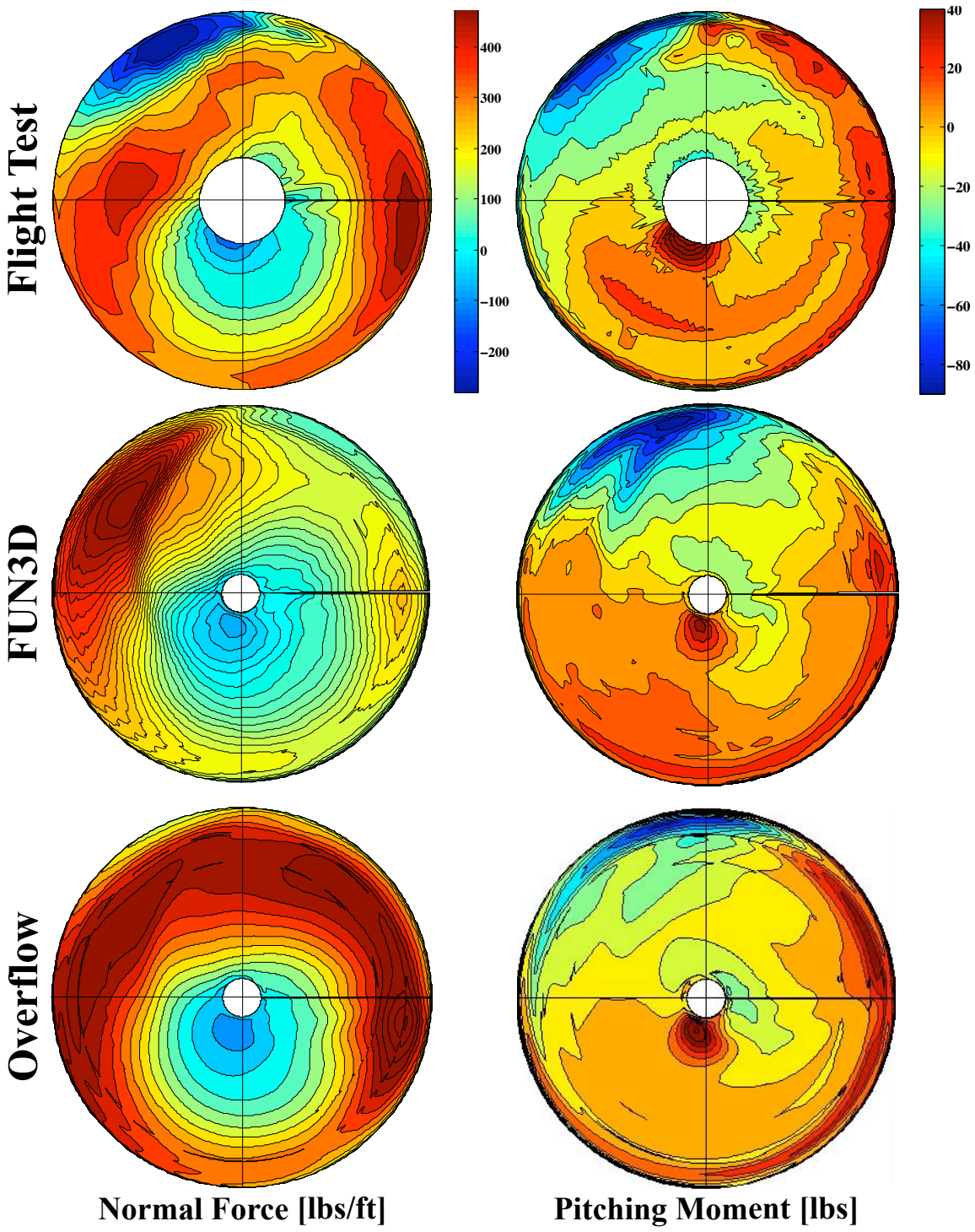


Figure 85: UH60A case 8534 polar plots of the normal force [lbs/ft] and the mean removed pitching moment [lbs] for the rigid motion cases compared to flight test data and structured CFD, structured predictions courtesy of reference [3], freestream flow from left to right

additional positive lift can also be seen in the Cp slices in Figures 86 and 87. Although the slices for $\psi = 0^o$ and $\psi = 90^o$ indicate that the lift is the same or less than the flight test data, the slices for $\psi = 180^o$ have higher suction peaks than the flight test data, indicating that there is more lift predicted in this region than measured in the flight test data. The confirmation of this elastic effect will be shown in the next section on elastic deformation.

Comparison to the structured CFD case yields many of the same conclusions. The normal force predicted near the tip of the blade is higher than flight test data, although the prediction on the advancing blade is higher than for the unstructured CFD case. Similarly, the pitching moment results are more accurate closer to the root of the blade where the elastic deformation has less impact on the results. The high positive moment in the reverse flow region is captured by both unstructured and structured CFD methods, However, the structured CFD case predicts higher magnitudes in this region. The results near the tip of the blade tend to over-predict the moments in the unstructured CFD case, and under-predict them in the structured CFD case.

The reverse flow region, the blue region in the normal force plot in Figure 85 bordering the root cut out, is captured. Figure 89 illustrates the local velocity surrounding an airfoil slice in this region and confirms that reverse flow is modeled at this location. Figure 88 shows the iso-vorticity surface and the vorticity streamlines of the rotor wake after three revolutions. The wake is propagating behind the rotor in the direction of the incident freestream velocity. There is wake roll-up on the advancing side of the rotor, as well as root vortex roll-up. After three revolutions the physical wake has 12 helical loops, but numerical dissipation in the CFD code only preserves the loops located within one rotor diameter of the hub.

The general effect of using rigid blade motion in this case is that the advantages of using CFD to model the flow are not fully realized. This is because the rigid model neglects the elastic deformation, which is an important aspect of the real system. Thus the more accurate aerodynamic computations are being applied to effectively a different model, reducing the

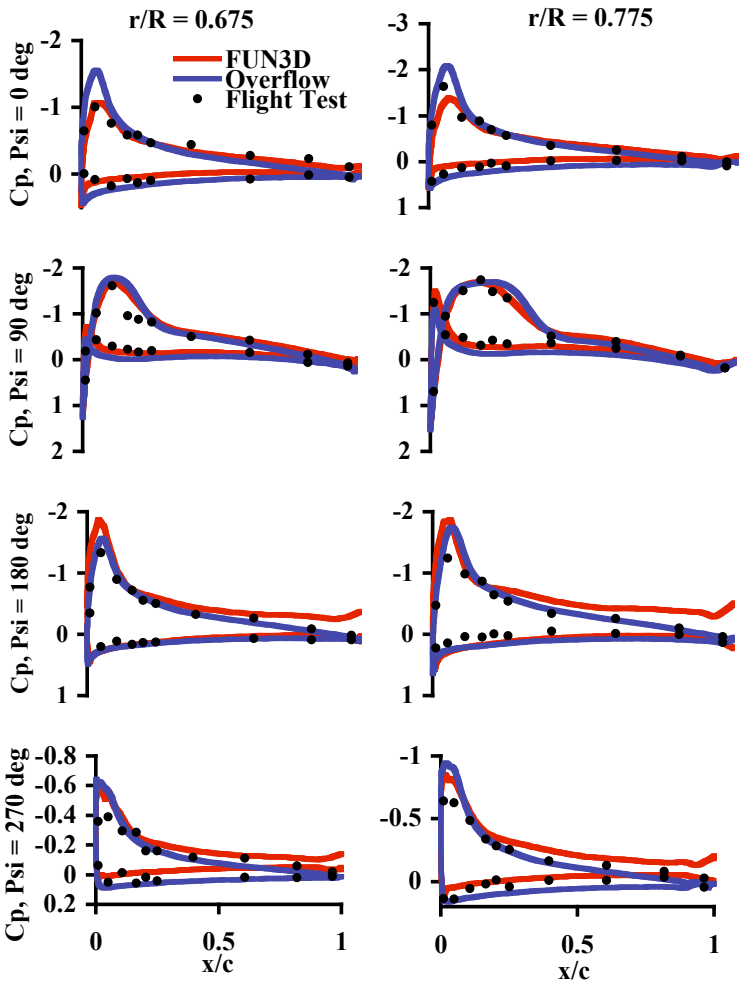


Figure 86: UH60A case 8534 blade Cp distributions for selected slices of the rigid motion case compared with flight test data and structured CFD, structured predictions courtesy of reference [3]

accuracy of the predictions. Modeling the elastic motion is required in order to fully-represent the blade motion; the elastic results and comparisons will be presented in the following sections.

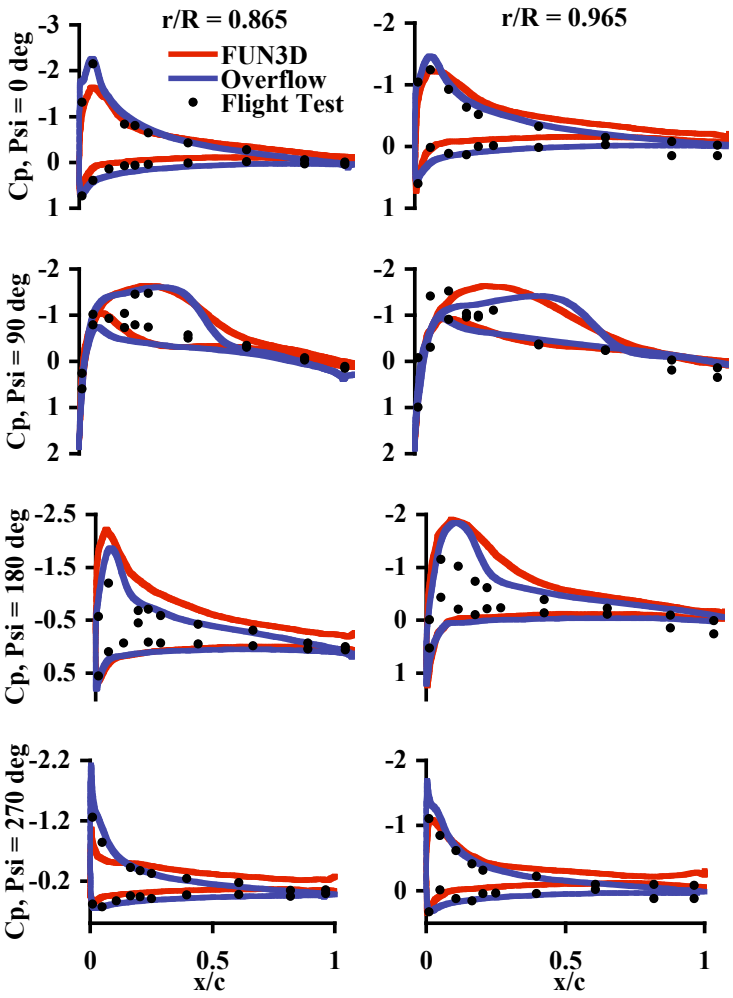


Figure 87: UH60A case 8534 blade Cp distributions for selected slices of the rigid motion case compared with flight test data and structured CFD, structured predictions courtesy of reference [3]

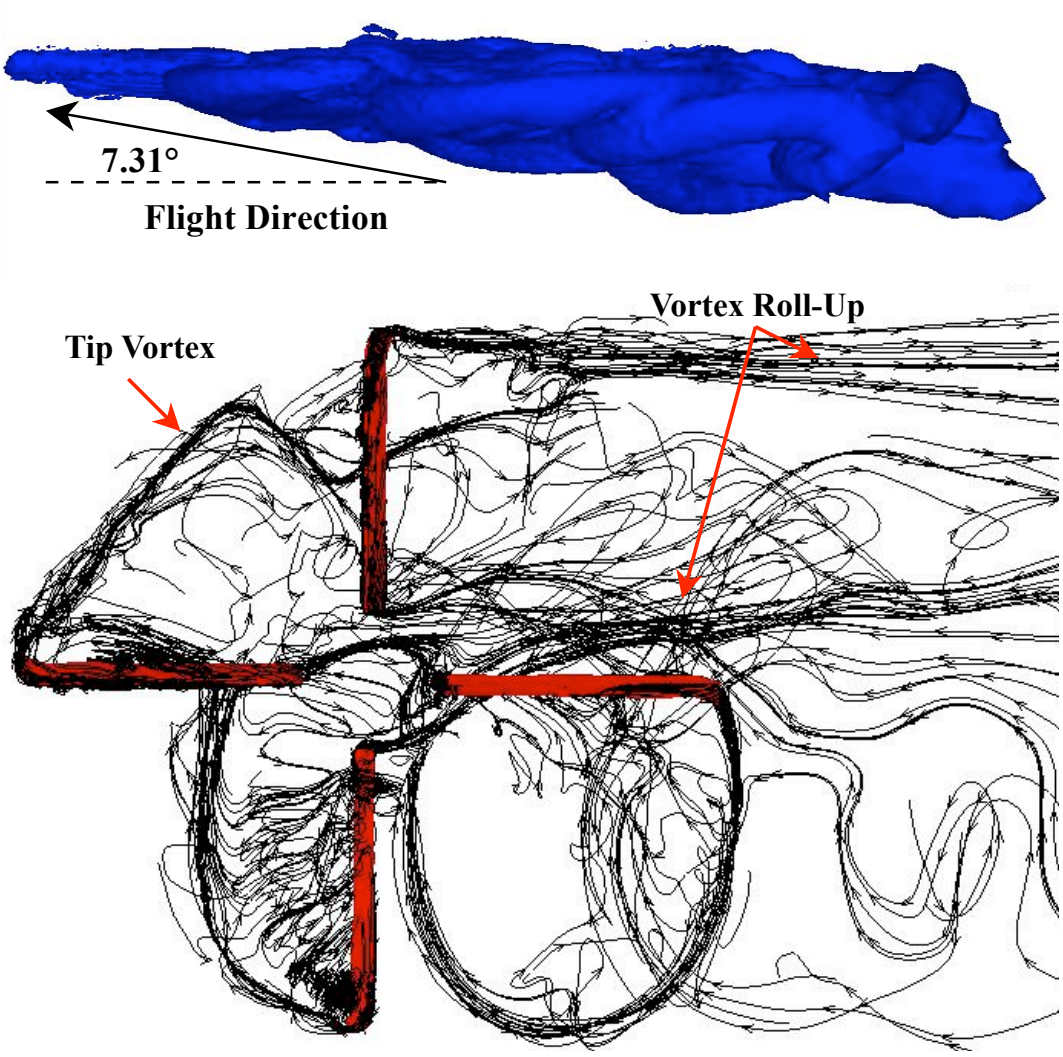


Figure 88: UH60A case 8534 vorticity plots of the wake geometry for the rigid motion case

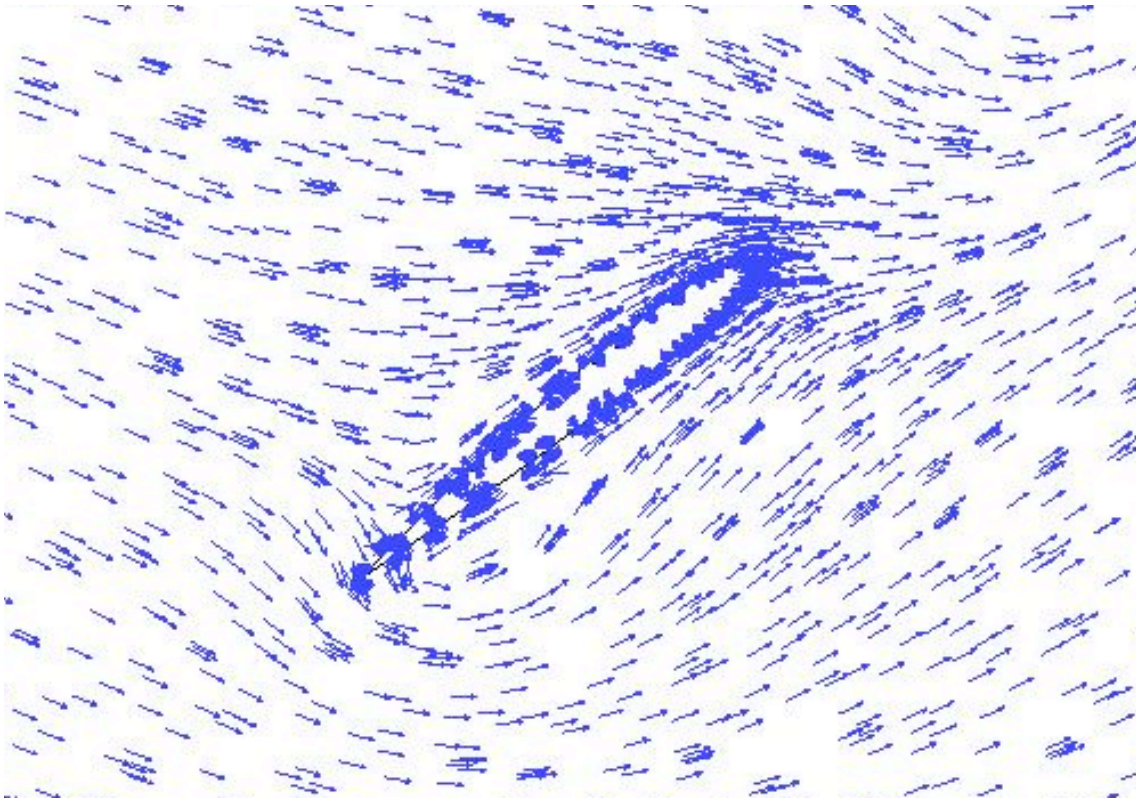


Figure 89: UH60A case 8534 reverse flow velocity vectors in the blade section frame at r=22.5%R and $\psi = 270^o$

7.2 UH60A Prescribed Elastic Motion Case 8534

The prescribed elastic motion case uses the same interface as the CFD-CSD coupling, but instead of passing unconverged data between two codes, a set of converged elastic motions obtained from coupling DYMORE with OVERFLOW are used as prescribed deflections. This motion data was obtained from a case run by reference [3]. Comparison is made to the corresponding structured CFD data computed using OVERFLOW for the same motion. The unstructured and structured CFD solver settings are discussed in sections 4.1.3 and 4.1.4, the grid differences are discussed in section 4.1.5. Since the prescribed data was obtained from a converged OVERFLOW-DYMORE coupling, the prescribed motion case, when using OVERFLOW as the CFD code, is equivalent to a fully-converged OVERFLOW-DYMORE case, whereas using the same data in the unstructured CFD code does not result in a converged CFD-CSD coupling case, but should result in loads comparable to the OVERFLOW simulation. Thus, it is expected that the structured code results may be closer to the experimental data than for the unstructured CFD code results. As expected, the results illustrated in Figures 90, 91, and 92 for both cases, although not identical, yield results that are close to the experimental data, where the phase of the data is similar for both cases. The magnitude of the normal force data do show some slight differences, which may be attributable to the differences in the scheme and grids. Unlike the prior rigid blade simulations, the negative lift region that occurs at 135^o near the tip of the blade is captured, confirming that it is a phenomena of the elastic blade motion. The negative lift peak occurs when the blade twists in a nose down direction, reducing the lift produced. The pressure coefficient data plotted against flight test data in Figures 93 and 94 confirm that there is an under-prediction of lift at $\psi = 180^o$. Although using prescribed elastic motion yields closer results when compared to the experimental data than the rigid case, the best results are obtained by using unstructured CFD-CSD coupling.

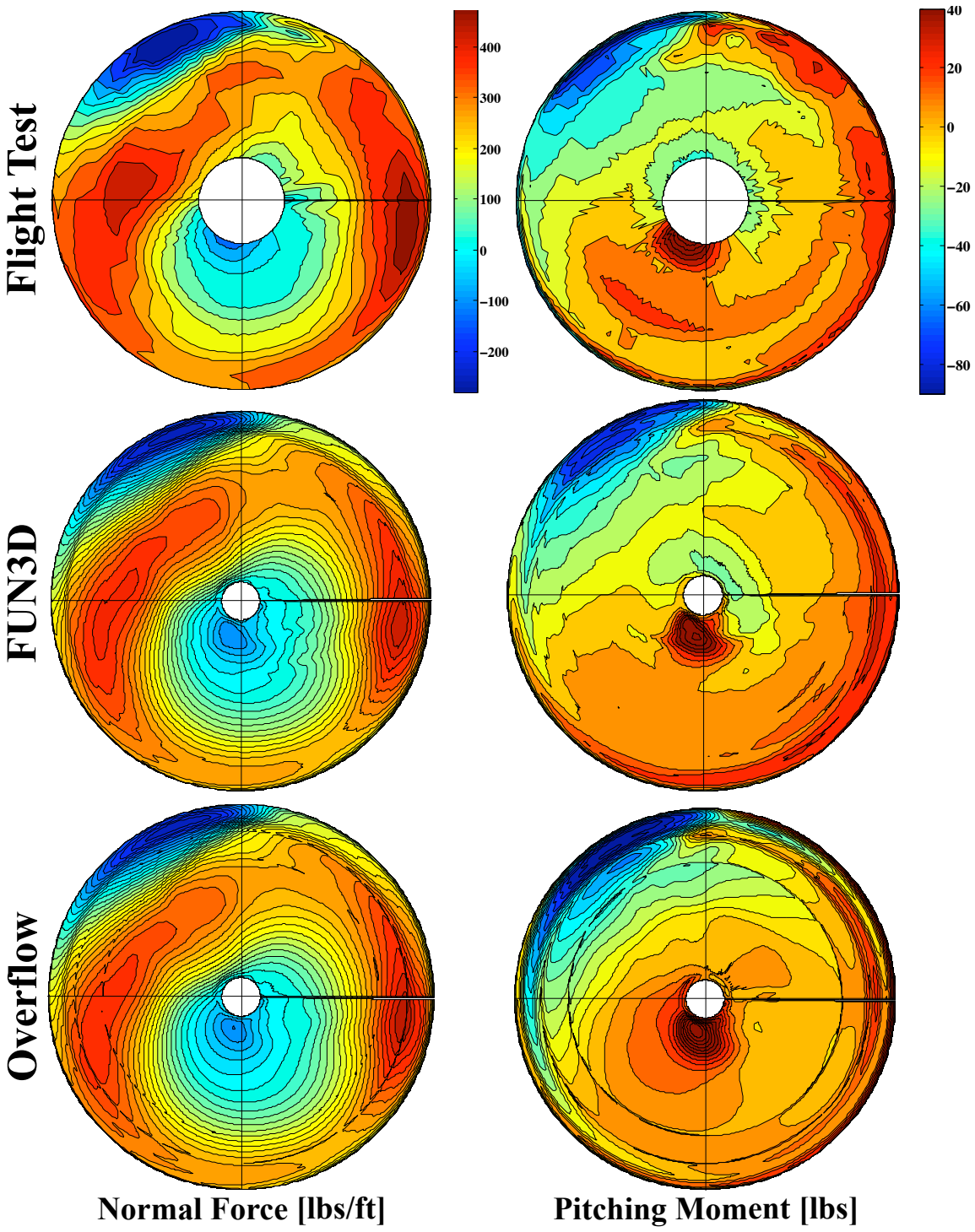


Figure 90: UH60A case 8534 polar plots of the normal force [lbs/ft] and the mean removed pitching moment [lbs] for the prescribed elastic motion case compared to flight test data and structured CFD, structured predictions courtesy of reference [3], freestream flow from left to right

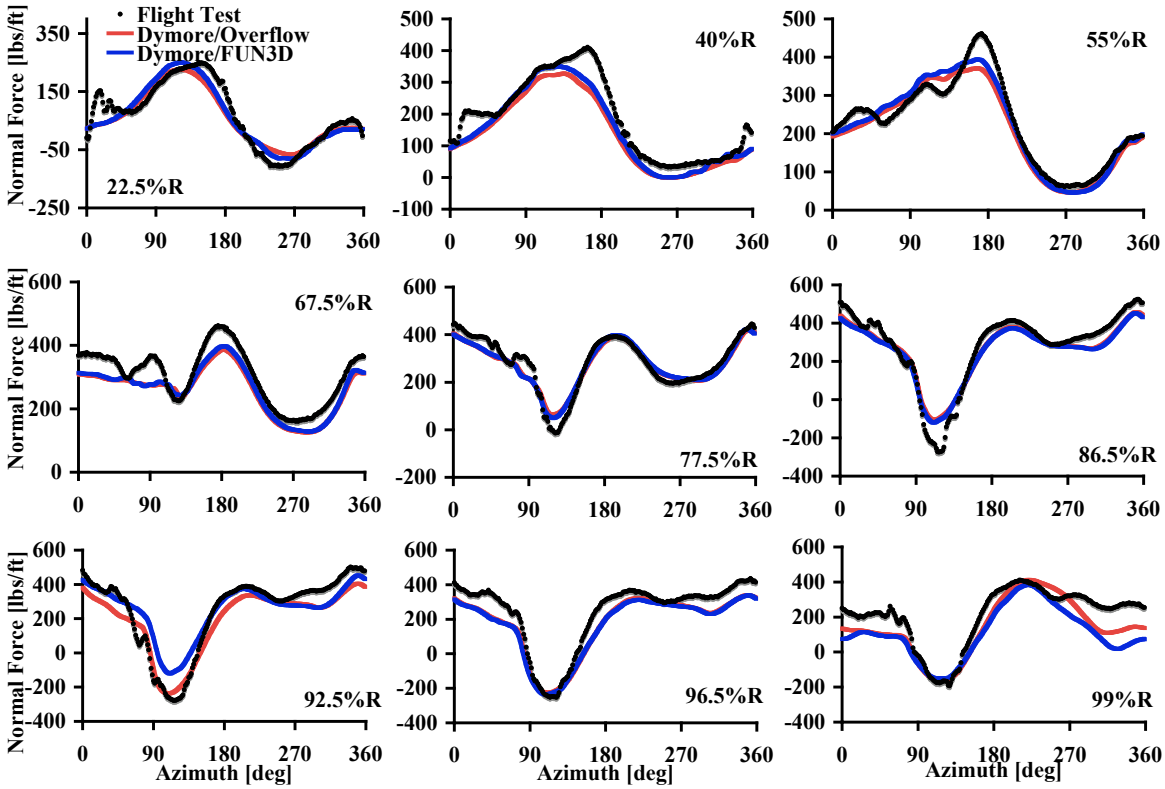


Figure 91: UH60A case 8534 normal force [lbs/ft] for the prescribed elastic motion case compared to flight test data and structured CFD, structured predictions courtesy of reference [3]

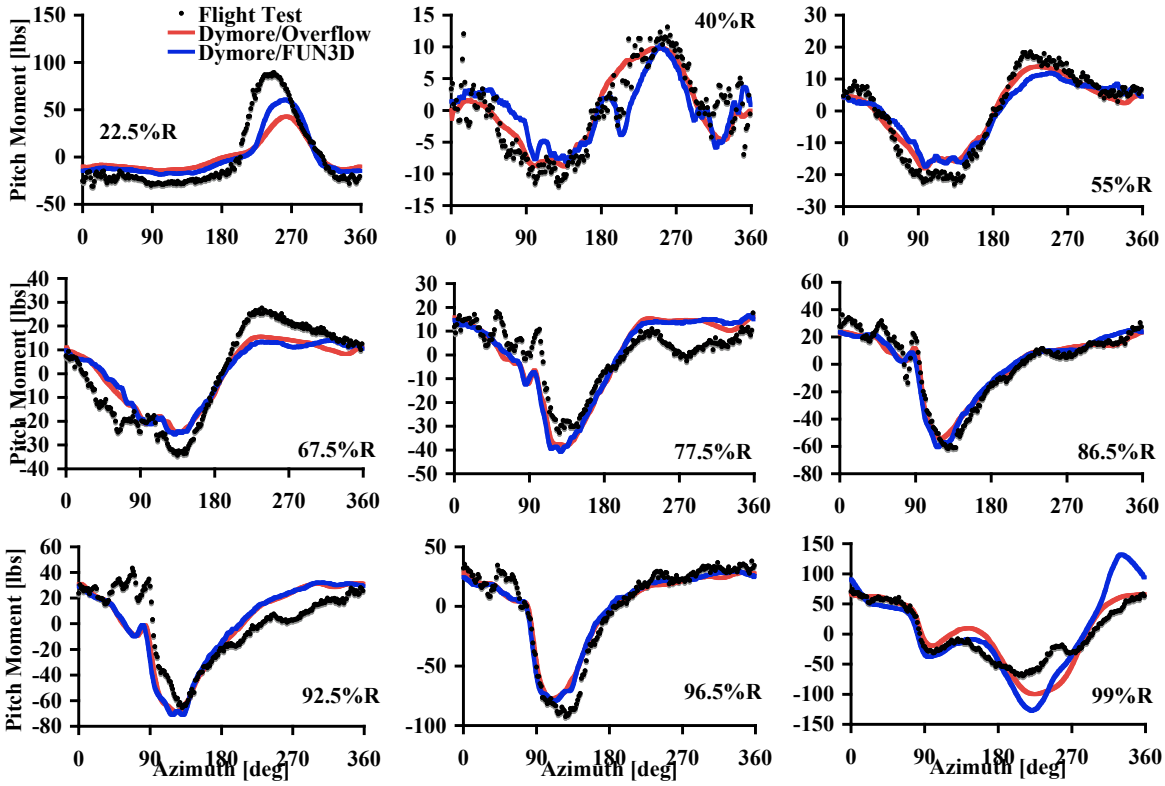


Figure 92: UH60A case 8534 pitching moment [lbs] for the prescribed elastic motion case compared to flight test data and structured CFD, structured predictions courtesy of reference [3]

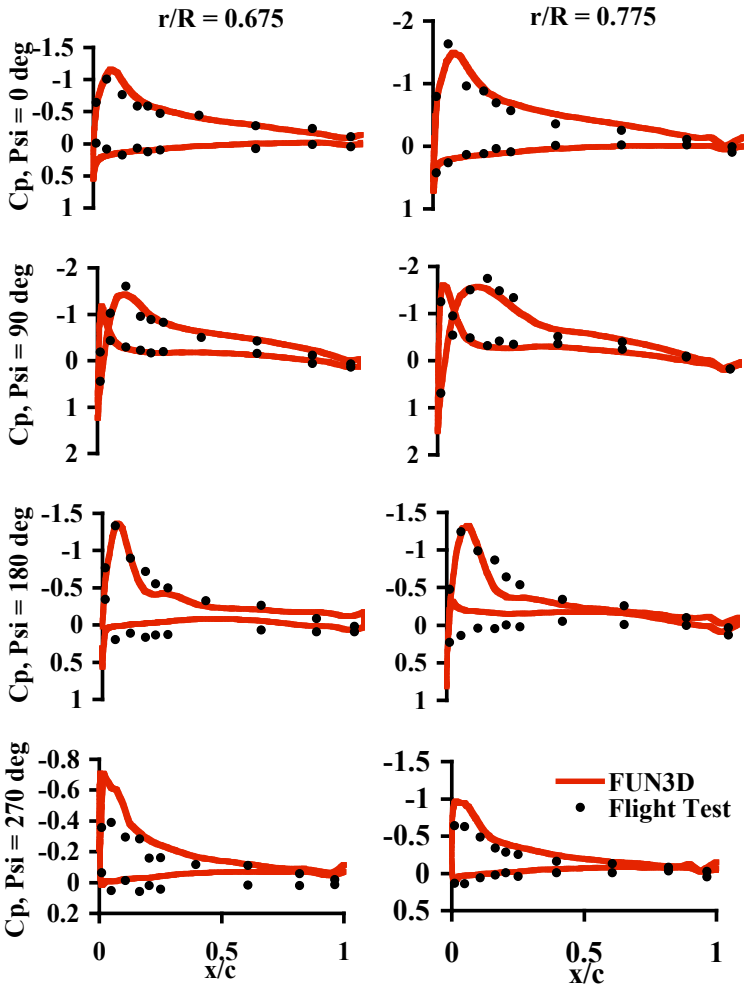


Figure 93: UH60A case 8534 blade Cp distributions for selected slices of the prescribed elastic motion case compared with flight test data

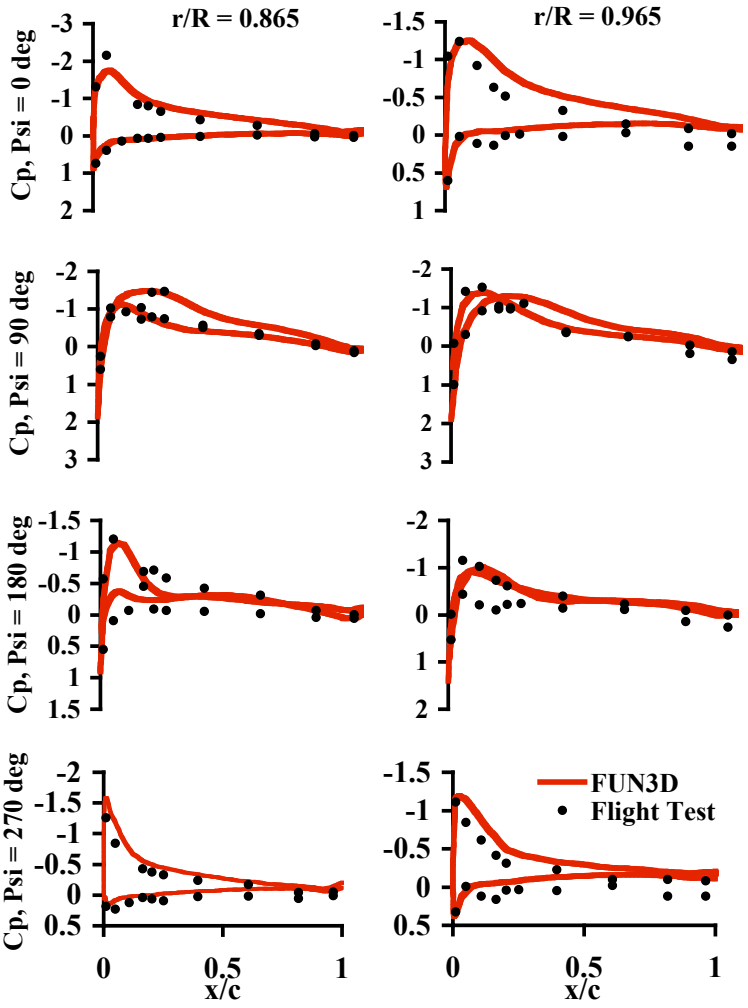


Figure 94: UH60A case 8534 blade Cp distributions for selected slices of the prescribed elastic motion case compared with flight test data

7.3 UH60A Computed Elastic Motion Case 8424

This is an elastic motion case, which passes data between DYMORE and FUN3D until a converged solution has been obtained. The DYMORE model has a resolution of 1^o per time step, and 81 non-uniformly spaced lifting line points clustered near the tip of the blade. The results are compared to coupled results run using an OVERFLOW-DYMORE coupling by reference [3]. Both methods use exactly the same blade surface geometry, though the grid surrounding this surface is different. The CFD solver settings are discussed in sections 4.1.3 and 4.1.4, the grid differences are discussed in section 4.1.5. The frame used to define the axes for the structural defections is illustrated in Figure 95.

These results are the most accurate when compared to the experimental data than any of the other cases run. The magnitude of the data is much closer to the experimental data and the phase is improved. The convergence of the coupling in Figure 96 approaches the target values as the coupling iterations progress, this is true of both the structured and the unstructured methods. The normal force and the pitching moment results, in Figures 97 to 99, have different characteristics when comparing the two CFD codes. The structured results, though the same as the unstructured results near the root of the blade, do not predict the vibratory loading on the advancing blade around the 67.5% radial station. While the unstructured results predict this vibratory loading, but do not damp out the vibrations as

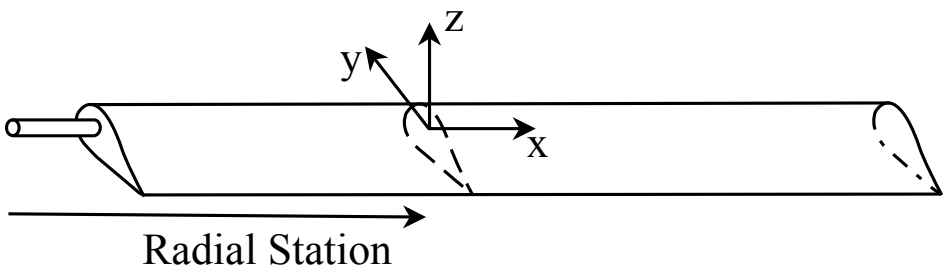


Figure 95: Illustration of the local frame coordinates used to define the motion data plotted in this section

quickly as they should on the surrounding stations. This point is further illustrated in Figures 103 and 104, where the harmonic decomposition of the normal forces and the pitching moments are plotted. The unstructured results have more higher harmonic content around 4/rev than the structured results. The phase predictions using the unstructured method are better, as are the magnitudes on the inboard blade stations, but the structured method has better magnitude predictions near the tip of the blade. Both methods under predict the loading at the sweep transition location. The pitching moment results are mixed. The high frequency vibrations in the unstructured results are an advantage in some locations and the damped out results predicted by the structured code are an advantage in others. The internal bending moments in Figures 100 to 102 also have varying results. The phase of the chord bending moment unstructured data is close to the flight test data whereas the structured results are out of phase. However, the magnitude of this unstructured data is much higher then the flight test data especially closer to the tip of the blade. The structured results have a much lower magnitude overall. The flap bending moment predictions are about the same when looking at both the structured or the unstructured results, neither method has an advantage over the other. The torsional moment predictions are missed by both the structured and the unstructured results. However, the unstructured data shows higher harmonic content near the advancing blade that is not present in the flight test data. It is not surprising that in this coupled problem the presence of higher harmonic content, illustrated in Figures 105, 106, and 107, is a trend amongst the unstructured results.

The structural deflections show many of the same characteristics as the airload data. Figures 108 to 113 compare the three translations and three rotations of each of the 81 lifting line stations defined in the structural model for the structured and unstructured CFD methods. These figures contain carpet plots of both the harmonic decomposition and the delta deflection from the initial reference position of each station. Figure 108 represents the blade stretching along the span axis over one rotor revolution. This motion is dominated by the centripetal forces on the rotating blade and the out of airfoil plane aerodynamic forces.

Neither of these contributions is significantly affected by the difference in the CFD solver. The results for both methods show many of the same characteristics in both the value and the harmonic content. Figure 109 represents the translation of each blade section in the lead-lag direction. This motion is related to the aerodynamic drag forces and shows some higher harmonic content that is not present in the structured results. This harmonic content is at the same frequency as the harmonic content of the chord bending moment. Figure 110 represents the translation of the airfoil section in the flap direction. This motion is influenced by the normal forces. However, there is no higher harmonic content present in this data, both methods appear to have predicted approximately the same motion. The blade twist is represented in Figure 111. Only the first harmonic is present in this data, which is not unexpected since there is no higher harmonic control present in the trim model. However, each CFD method predicts variations in the magnitude of this motion. Figure 112 represents the rotation of each airfoil section about the y-axis which points from the trailing edge to the leading edge of the blade. This motion does indicate some higher harmonic content near the tip of the blade and is related to the flap bending moment. Figure 113 represents the rotation of the blade section about the normal axis. This motion shows the most higher harmonic content and appears to be related to the changes in the chord bending moment. This should be the case since the chord bending moment is related to the same z-axis rotation seen in this figure. It is clear that the CFD-CSD methods predict more accurate airload results than other lower delity methods, as expected. While both the structured and unstructured methods predict overall the same airloads, as described using the carpet plots above, there is one exception in the chord bending moment. It is clear from Figure 105 that while both codes are predicting the harmonic content at the blade tip, the unstructured method does not damp out these harmonics inboard on the rotor blade. The frequency content is also informative, illustrating that the unstructured method's 5/rev content is dominant, which is not the case for the structured method and flight test. It is also observed that the chord bending moment is coupling with the torsional mode.

These differences can be traced to the differences inherent in the two code algorithms and the uncertainty in structural damping. For these runs, the structural damping is not known, and thus in Dymore there is no coefficient of damping added to the structural dynamics computation. However, the central difference schemes in OVERFLOW require that the user add numerical damping for stability, even when different dissipation schemes are utilized. FUN3D's numerical algorithms however do not require the addition of numerical damping, as the damping is inherent in the scheme. This is true overall for 2^{nd} order, implicit, upwind schemes and is shown through the amplification factor $|G|^2 = 1-4\sigma\left(1-\sigma\right)^2\left(2-\sigma\right)\sin^4\phi/2$ which specifies that the scheme is dissipative for $0 < \sigma < 2$. Further details of the stability analysis for this scheme may be found in reference [88]. Thus, it appears that FUN3D is more accurately simulating the structural response of the blade given that there has been no structural damping added. The addition of structural damping to the CSD method to investigate this is not straightforward. As the damping is not known, the scheme adds damping using the equation, $F = C\epsilon + \mu C\dot{\epsilon}$, where μ is the coefficient of damping. This damping is applied to the damping matrix in all three directions, thus the addition of damping in the chord bending will impact the other bending modes. Overall, the CFD-CSD methods predict more accurate airload results than other lower fidelity methods. However, the results obtained will vary depending on the CFD code chosen. Based on the results of the two codes presented here there can be variations in the magnitude, phase, and the frequency content of the results.

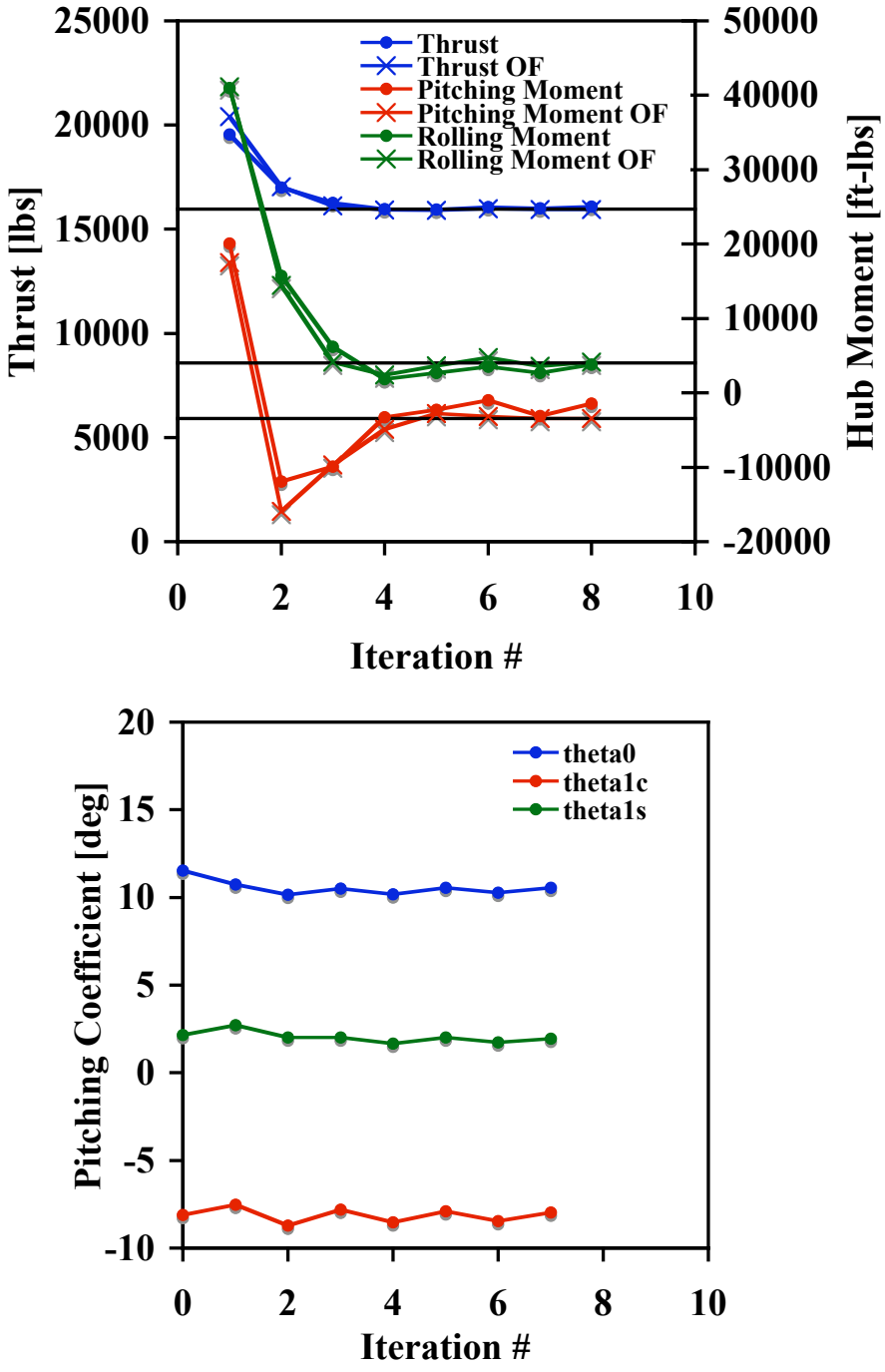


Figure 96: UH60A case 8424 convergence of the CFD-CSD coupling comparing both structured (labeled with OF) and unstructured methods, structured predictions courtesy of reference [3]

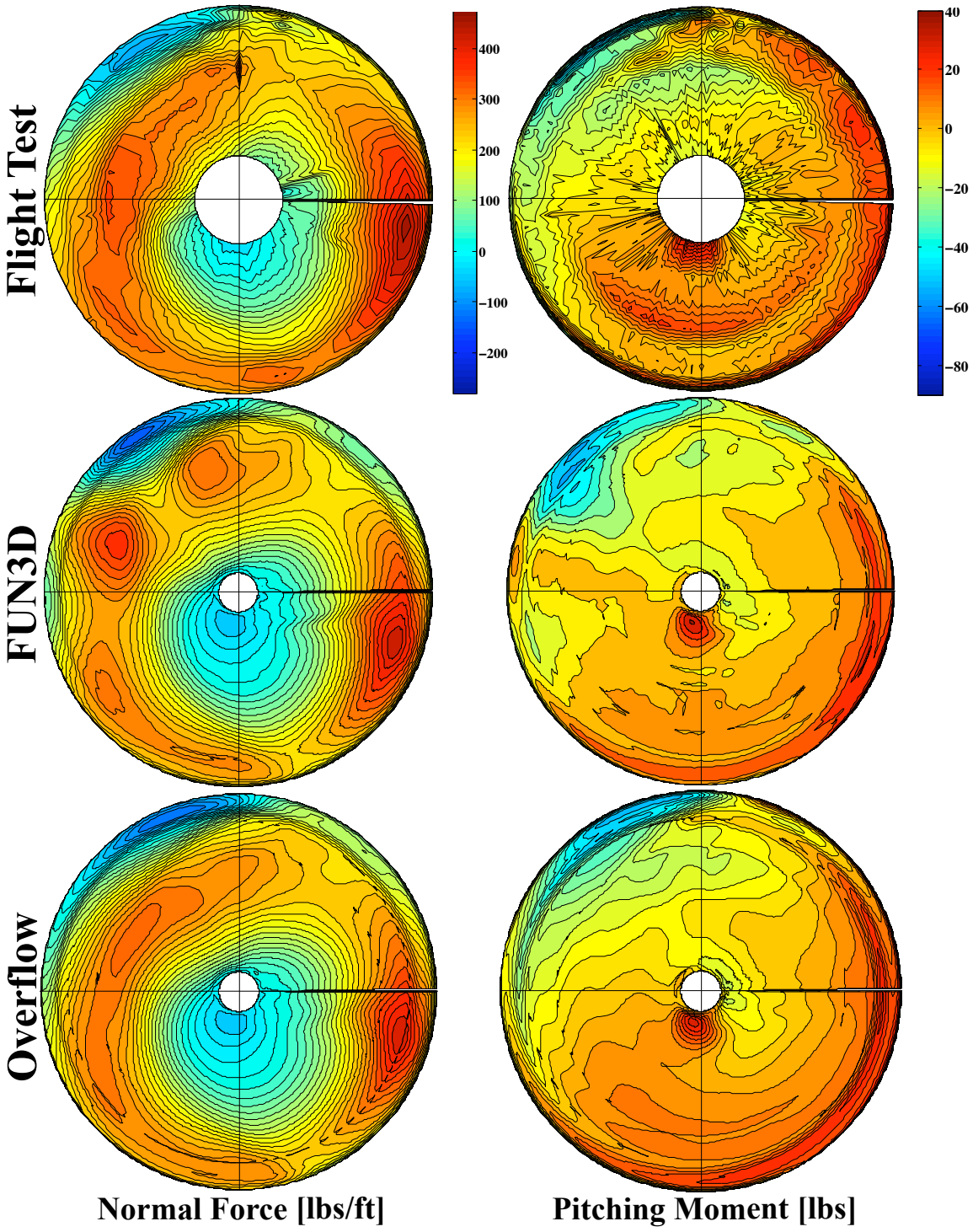


Figure 97: UH60A case 8424 polar plots of the normal force [lbs/ft] and the mean remove pitching moment [lbs] for the CSD coupled cases compared to the flight test data and structured CFD, structured predictions courtesy of reference [3], freestream flow from left to right

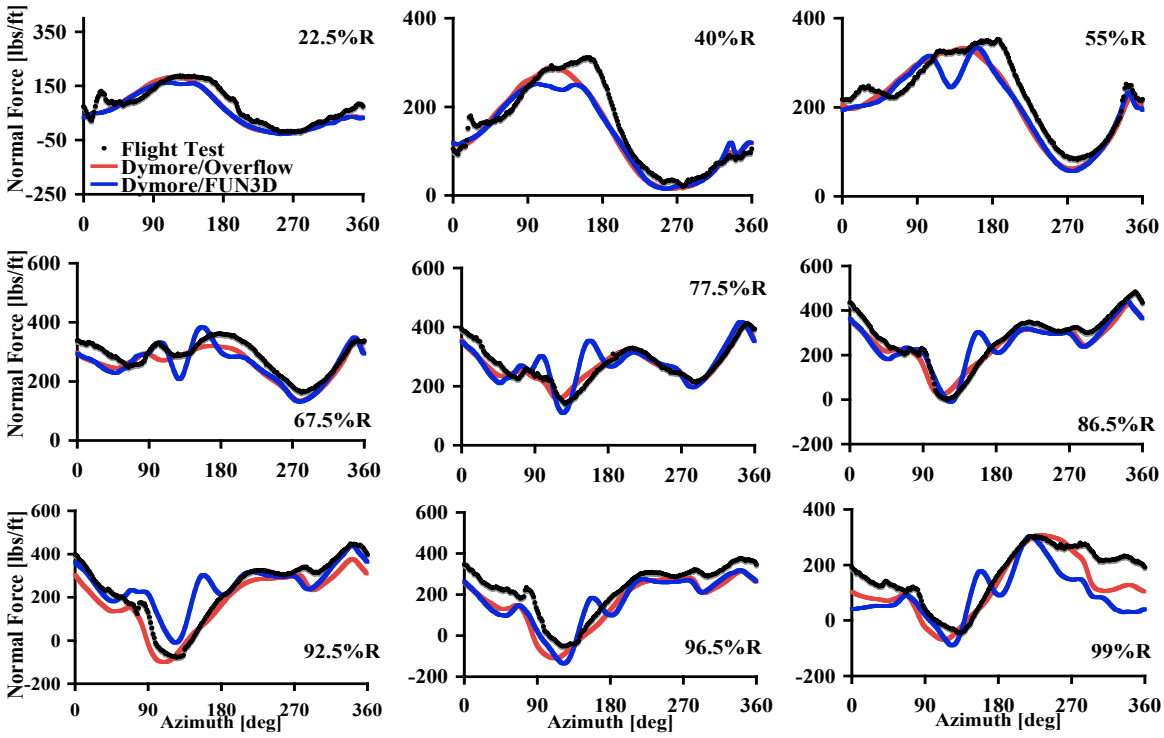


Figure 98: UH60A case 8424 comparison of normal force [lbs/ft] computations using CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

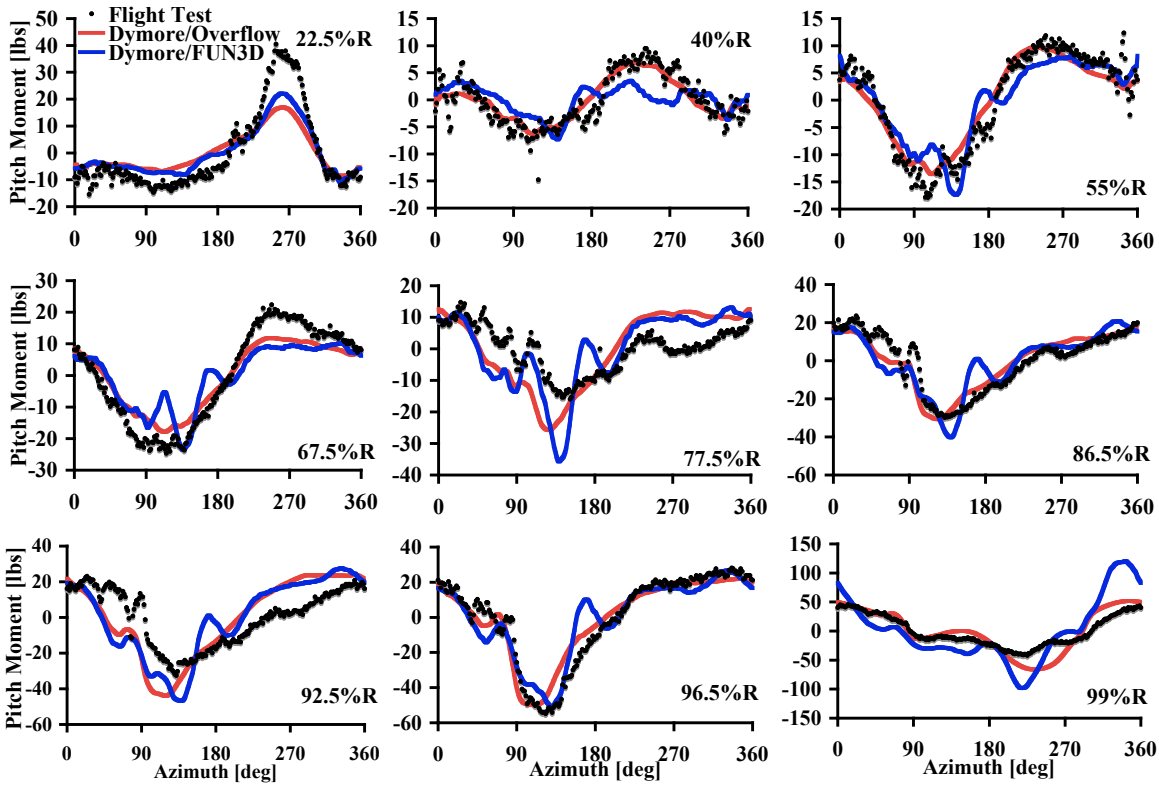


Figure 99: UH60A case 8424 comparison of mean removed pitching moment [lbs] computations using CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

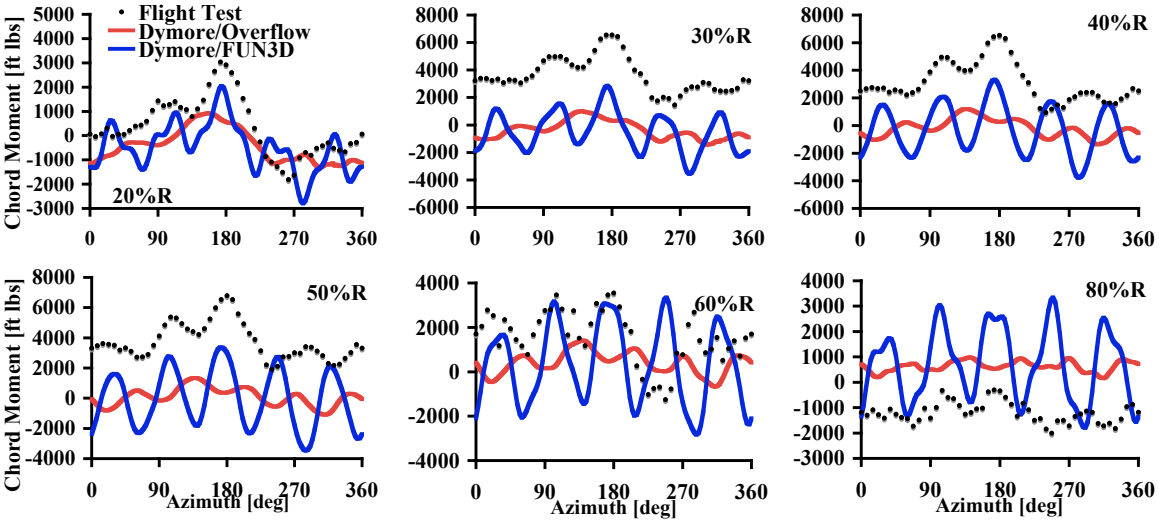


Figure 100: UH60A case 8424 comparison of chord bending moment [ft lbs] computations using CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

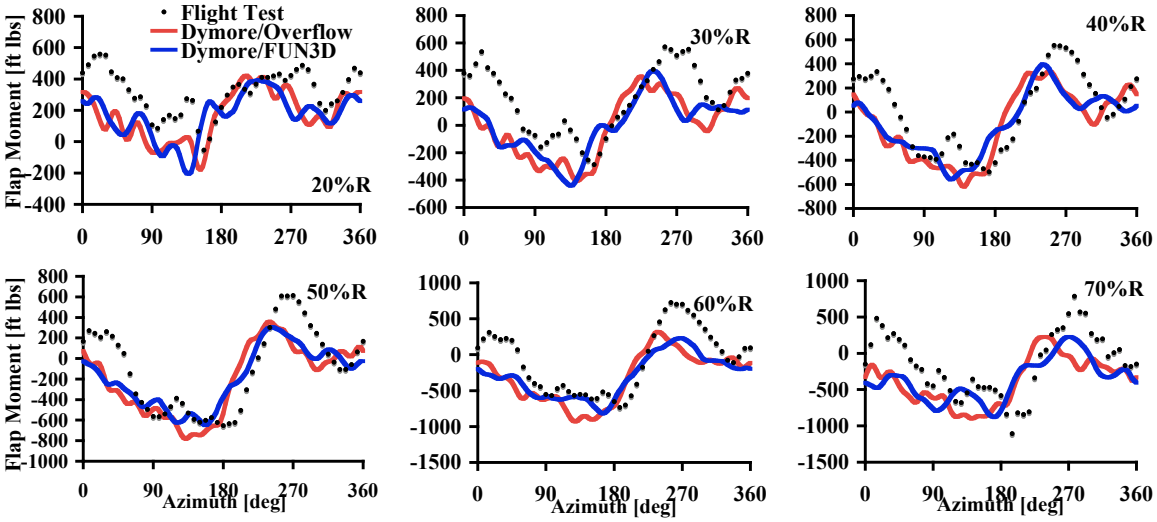


Figure 101: UH60A case 8424 comparison of flap bending moment [ft lbs] computations using CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

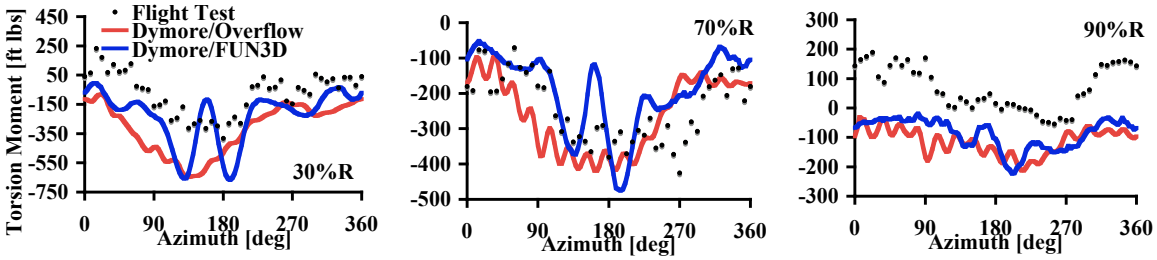


Figure 102: UH60A case 8424 comparison of torsional moment [ft lbs] computations using CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

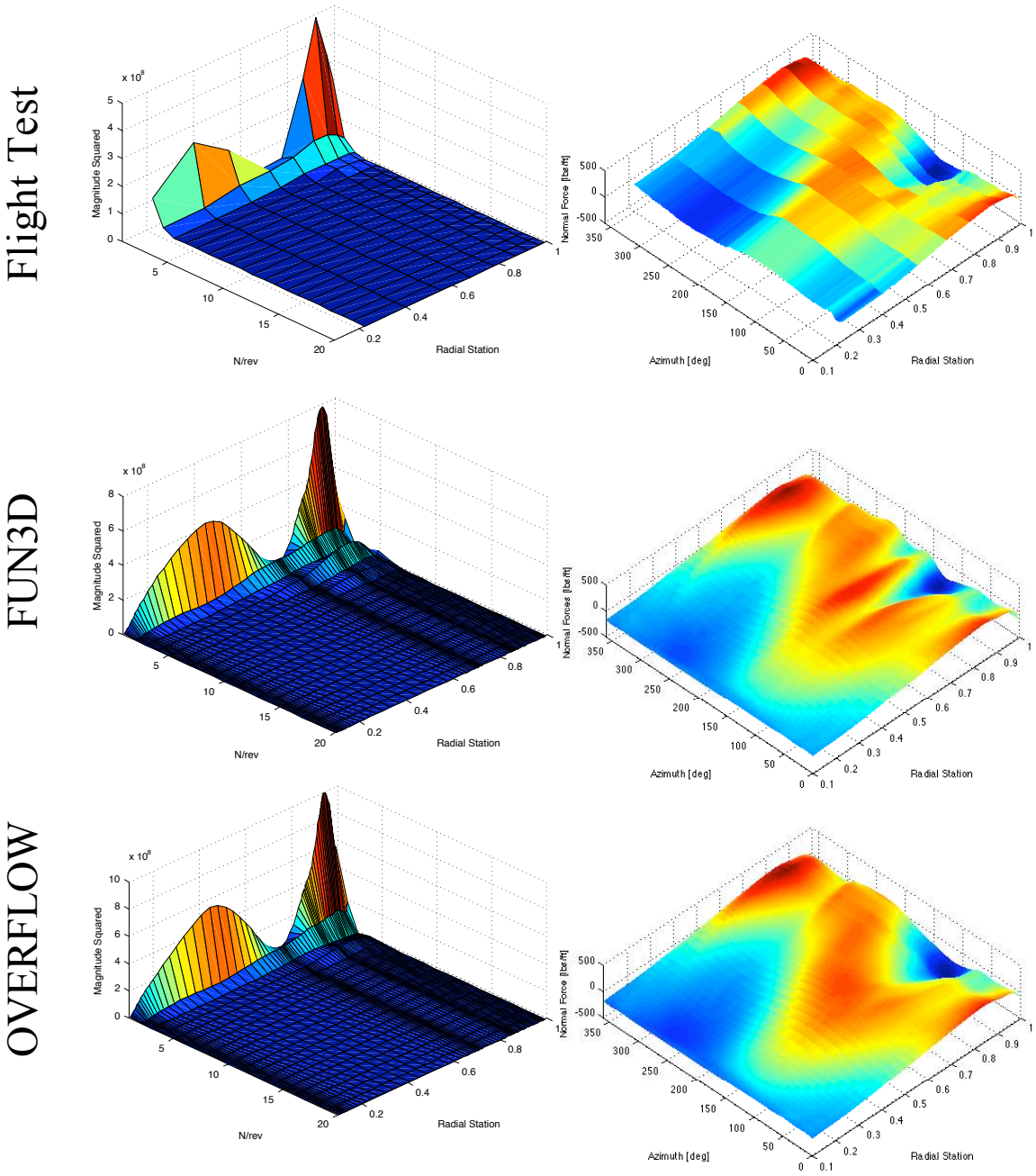


Figure 103: UH60A case 8424 harmonic decomposition and carpet plots of the normal forces [lbs/ft] comparing flight test data with CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

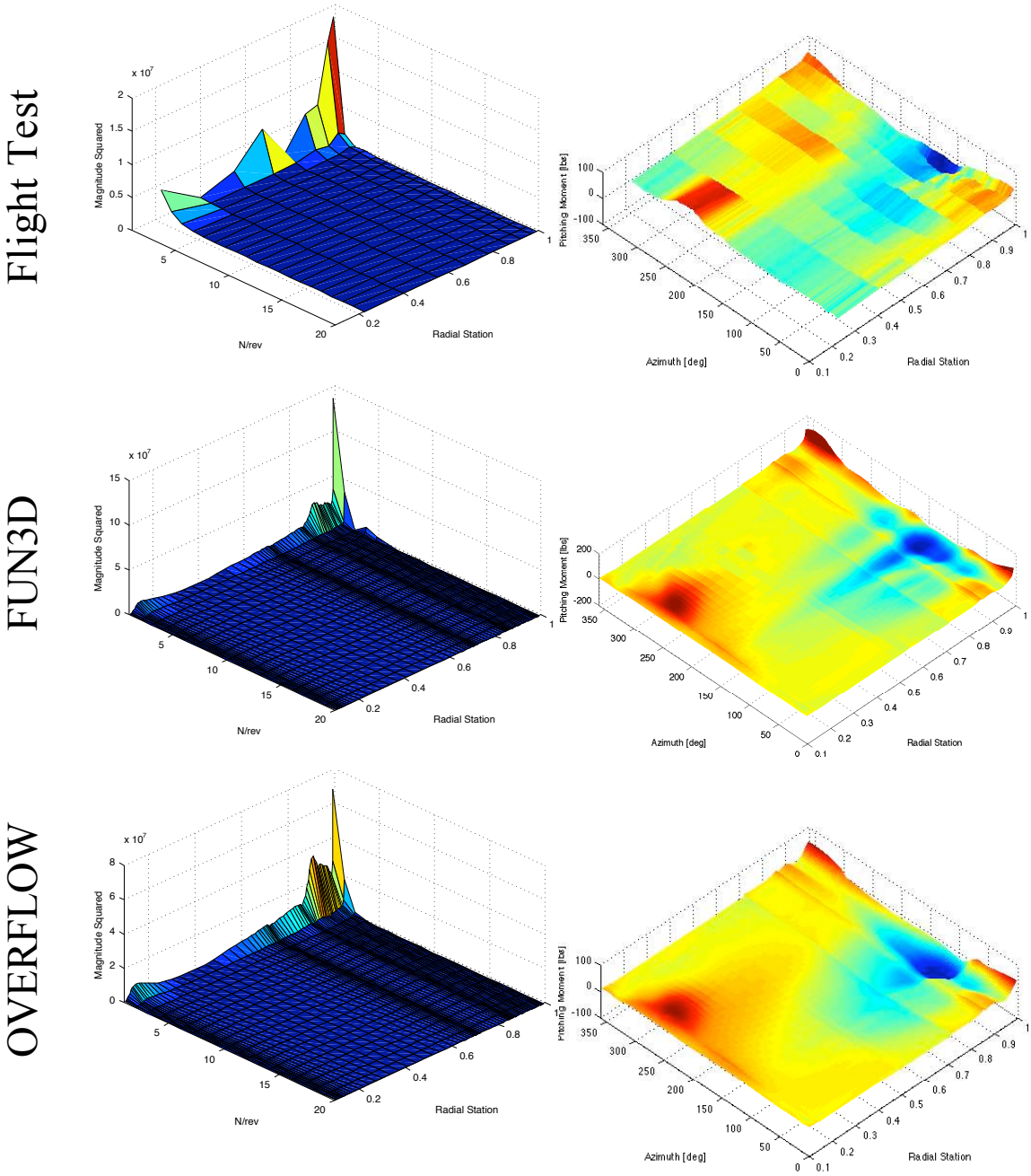


Figure 104: UH60A case 8424 harmonic decomposition and carpet plots of the pitching moment [lbs] comparing flight test data with CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

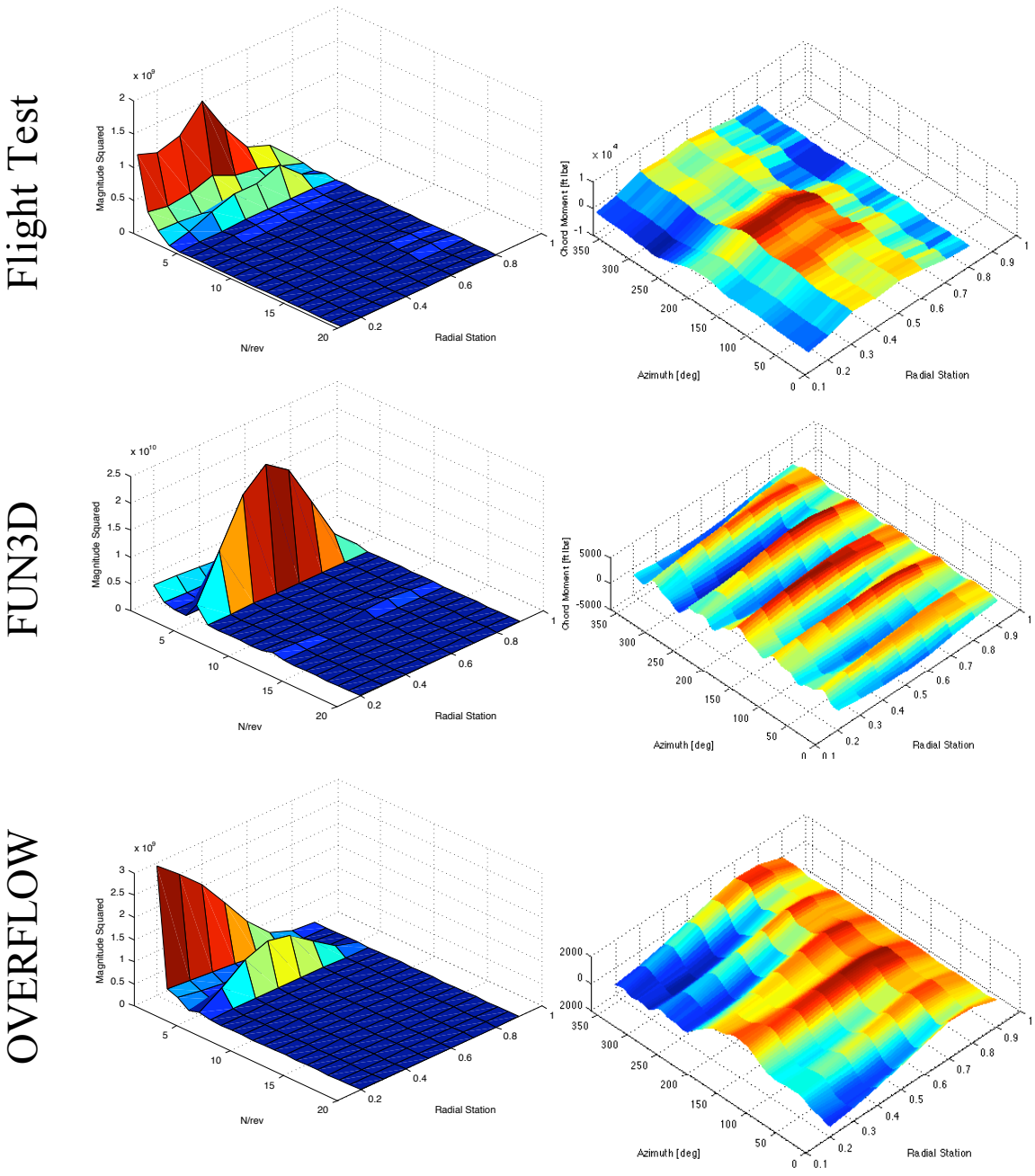


Figure 105: UH60A case 8424 harmonic decomposition and carpet plots of the chord bending moment [ft lbs] comparing flight test data with CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

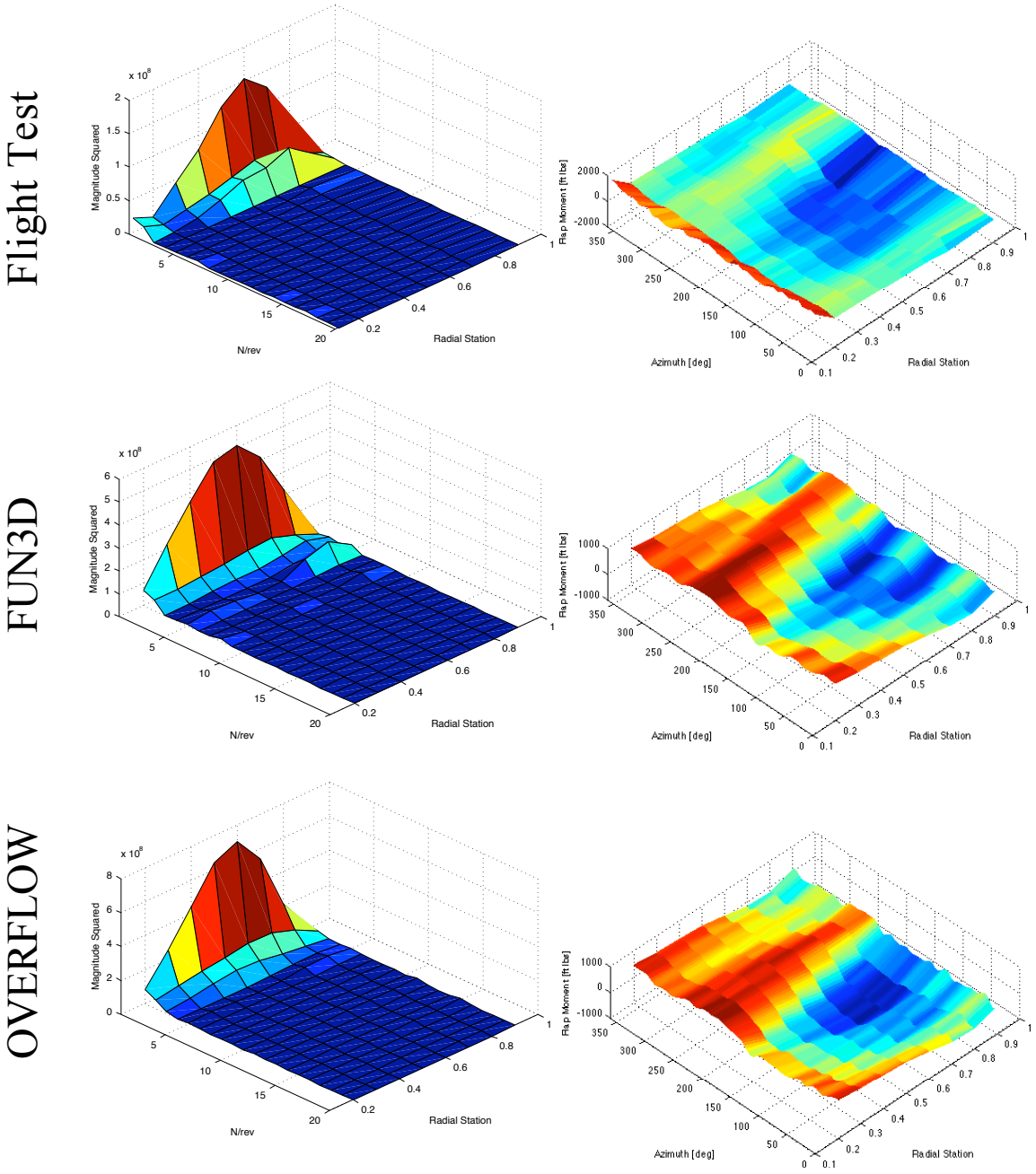


Figure 106: UH60A case 8424 harmonic decomposition and carpet plots of the flap bending moment [ft lbs] comparing flight test data with CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

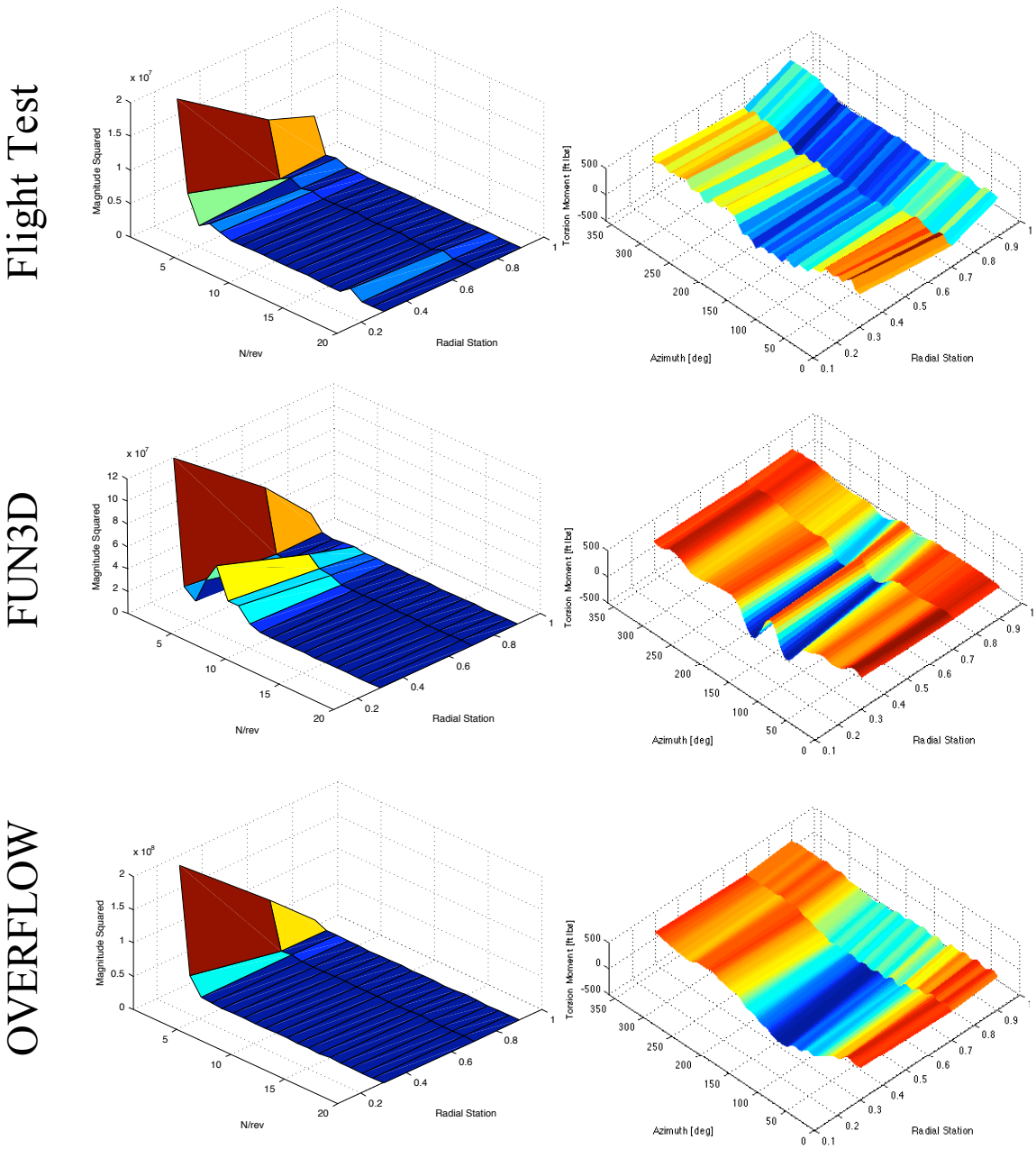


Figure 107: UH60A case 8424 harmonic decomposition and carpet plots of the torsional moment [ft lbs] comparing flight test data with CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

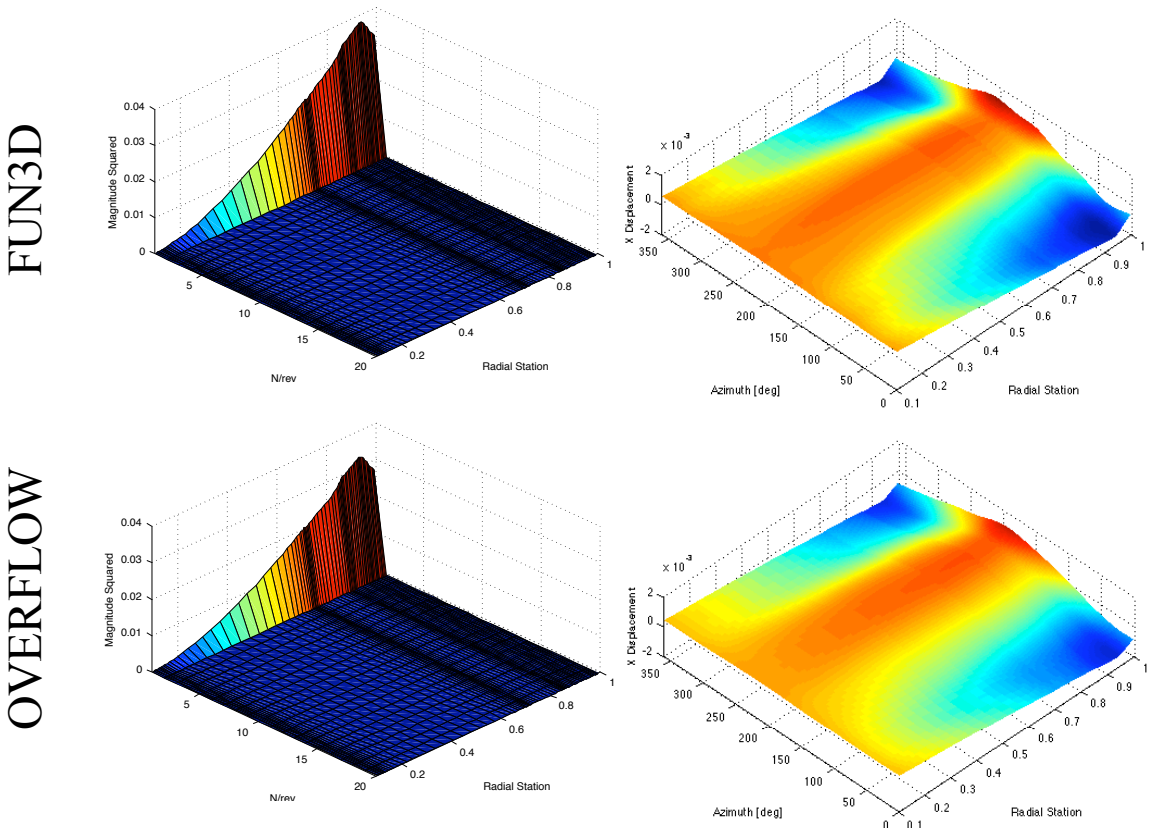


Figure 108: UH60A case 8424 harmonic decomposition and carpet plots of the sectional x displacement comparing CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

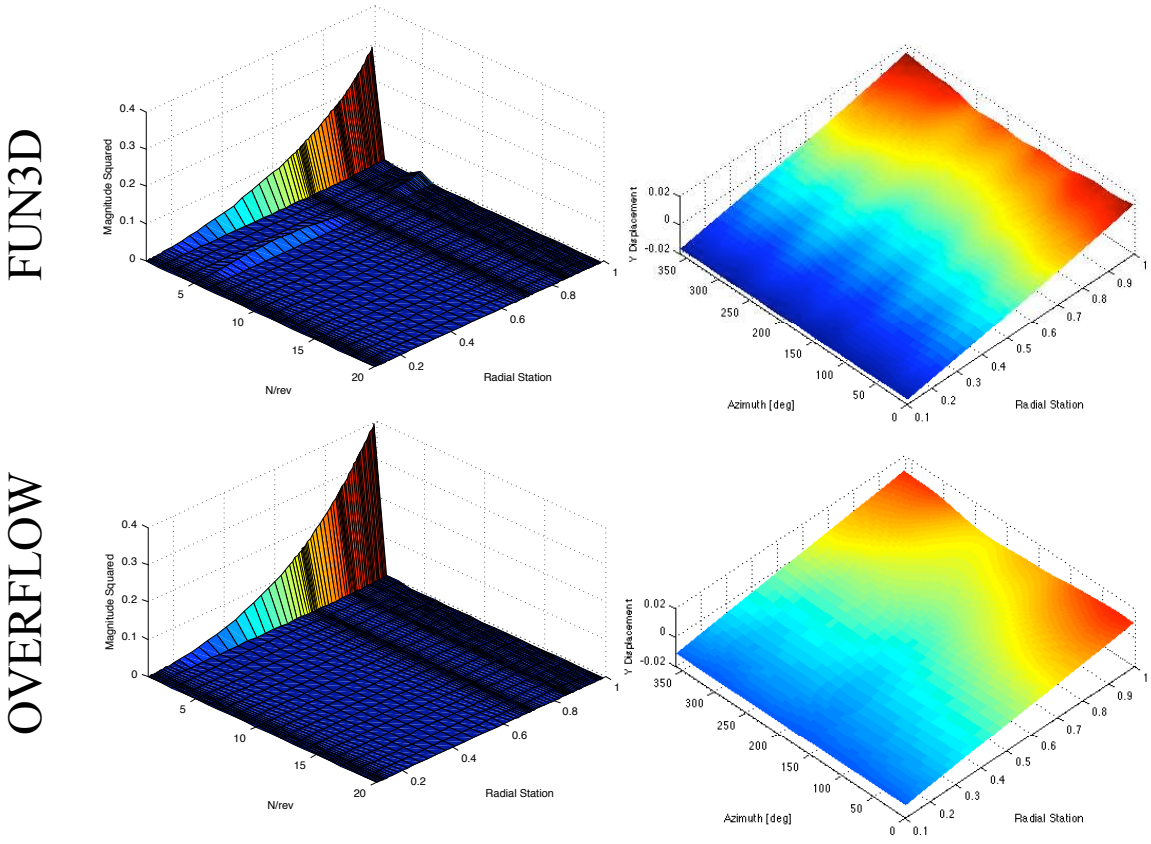


Figure 109: UH60A case 8424 harmonic decomposition and carpet plots of the sectional y displacement comparing CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

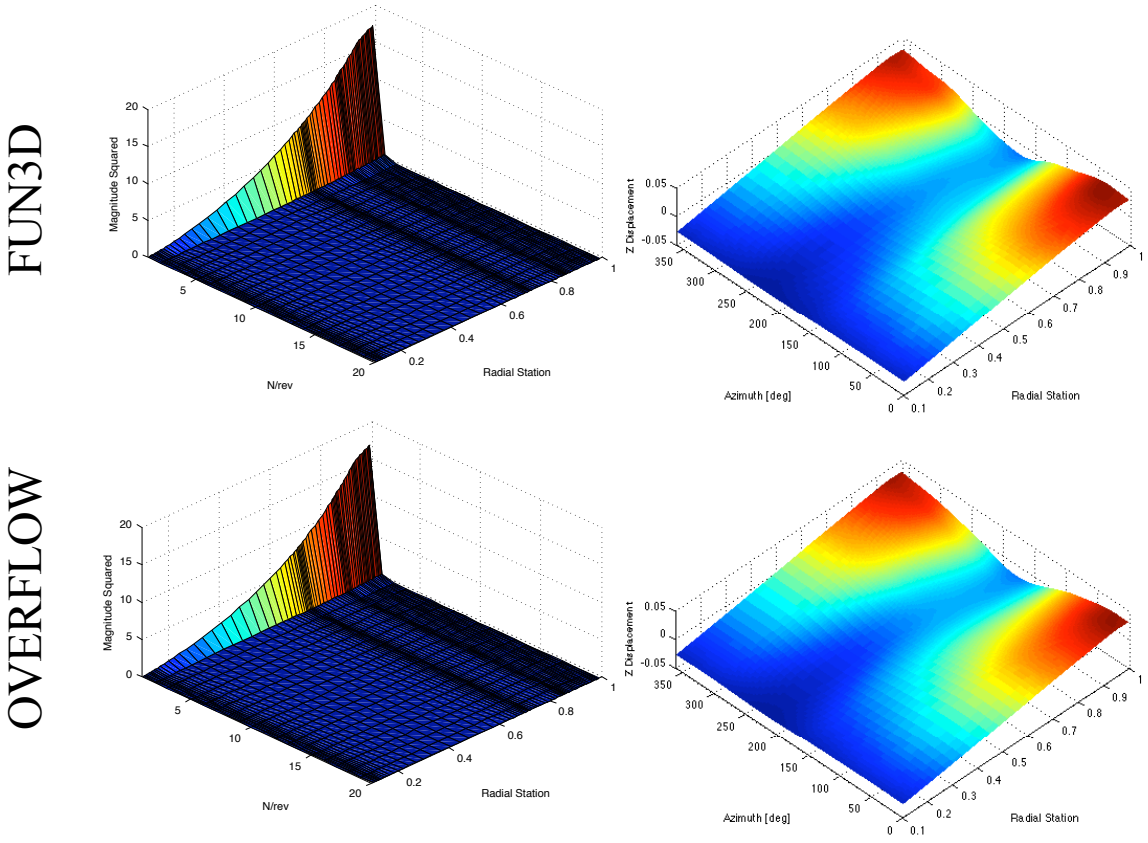


Figure 110: UH60A case 8424 harmonic decomposition and carpet plots of the sectional z displacement comparing CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

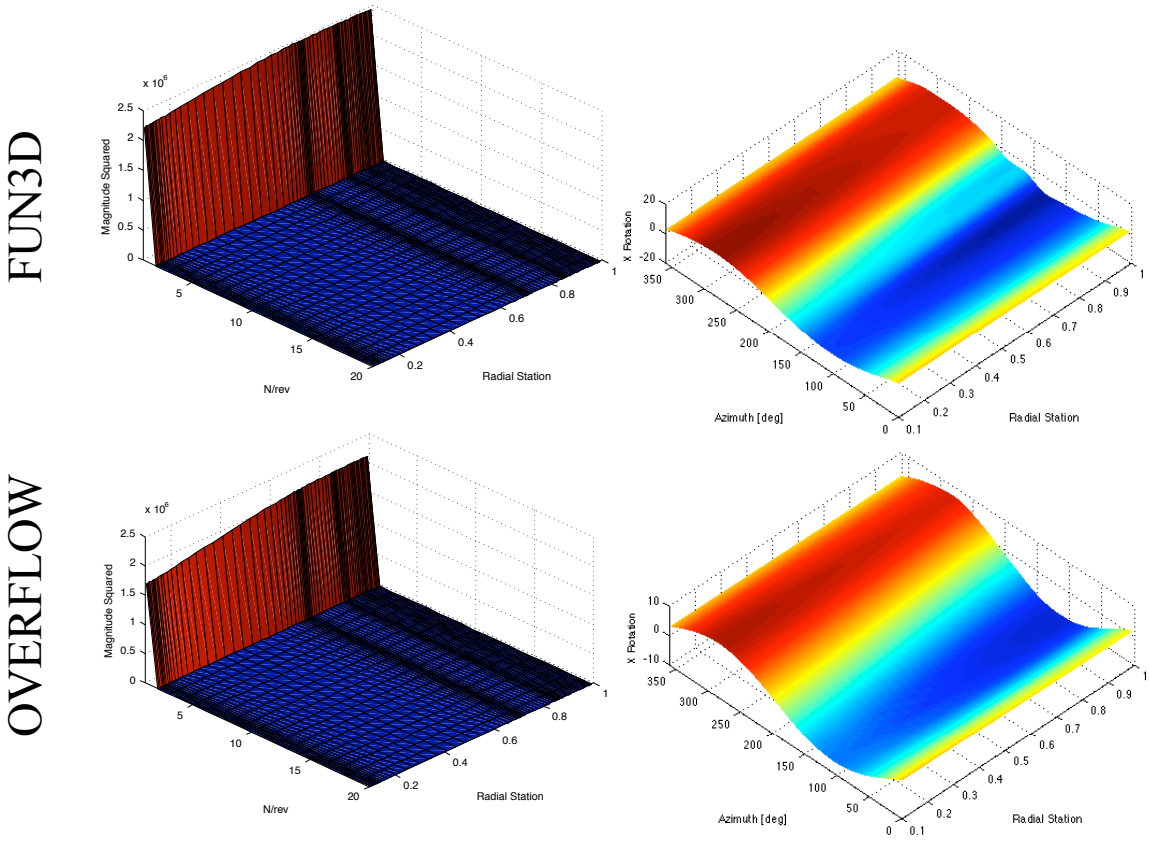


Figure 111: UH60A case 8424 harmonic decomposition and carpet plots of the sectional x rotation comparing CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

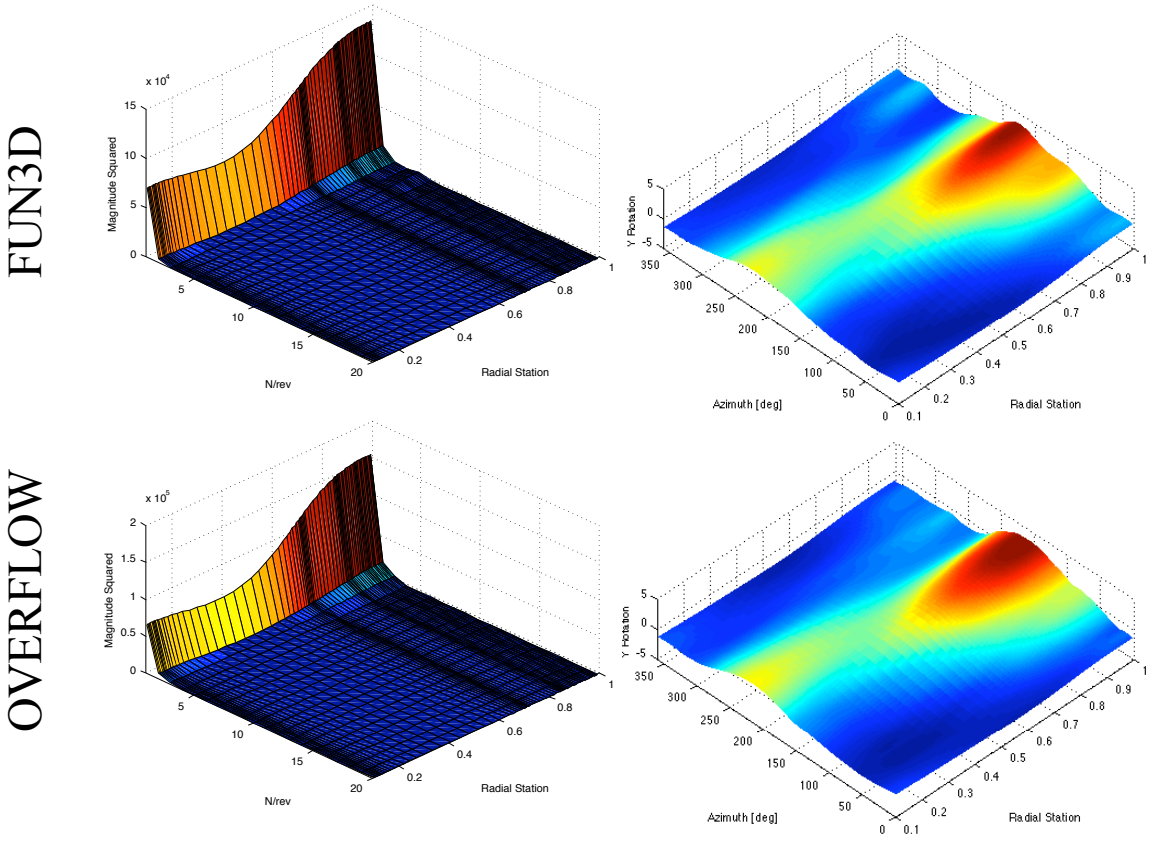


Figure 112: UH60A case 8424 harmonic decomposition and carpet plots of the sectional y rotation comparing CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

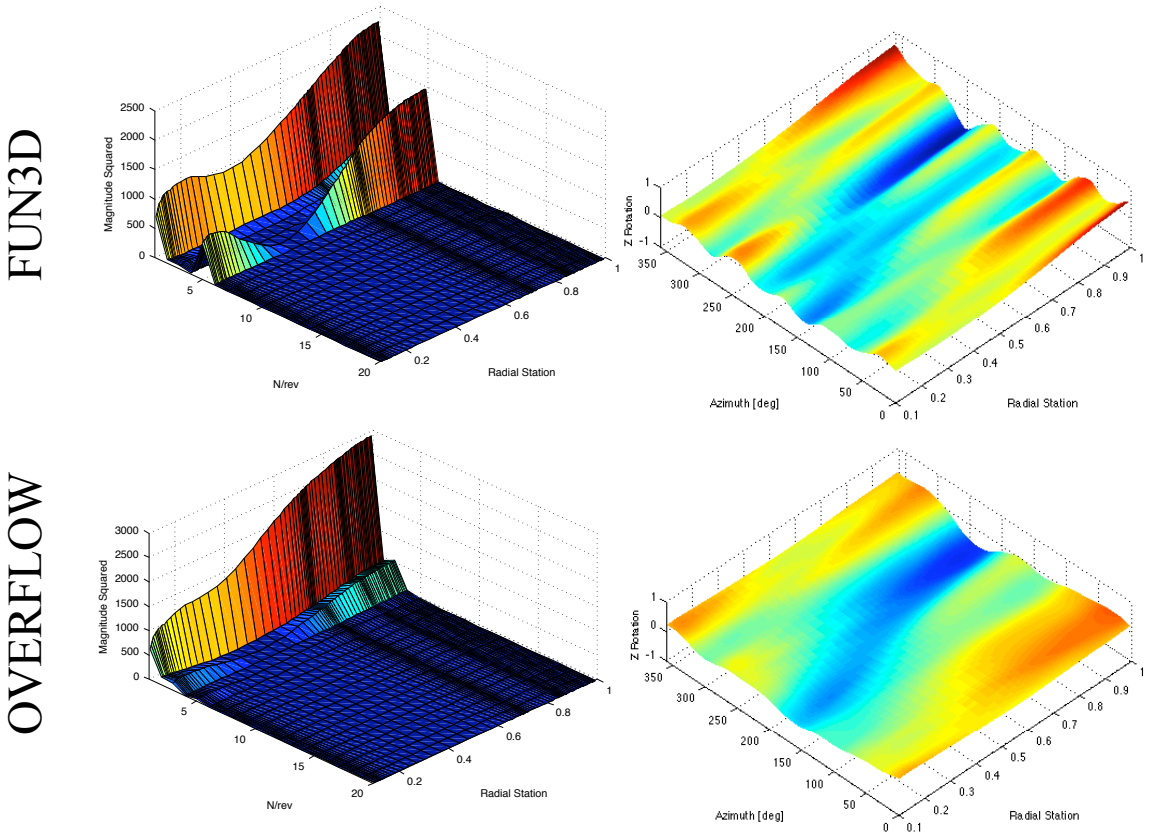


Figure 113: UH60A case 8424 harmonic decomposition and carpet plots of the sectional z rotation comparing CFD-CSD unstructured and structured coupling, structured predictions courtesy of reference [3]

CHAPTER 8

RESULTS: COMPARISONS

8.1 Coupling Method Analysis

In previous chapters, each of the types of coupling has been described independently of the other types of coupling. However, the different levels of fidelity used in each of the methods has a different impact on the accuracy of the solution. Figure 114 compares the normal force and pitching moment results for the prescribed elastic and the rigid cases. Qualitatively, the rigid results are the least accurate. The negative lift on the tip of the blade in the second quadrant is not modeled, the pitching moment on the tip of the blades is over-predicted in all quadrants, and at the root of the blade near the reverse flow region the pitching moment is under-predicted. This rigid model does not accurately represent the motion of the UH60A rotor under these flight conditions because the elastic motion is an important component of the motion. Figure 115 compares the elastic and rigid motion of the blades at selected locations. At the tip of the blades, the elastic twist is more significant, resulting in large changes in the predicted loads in this region. Specifically, the elastic twist induces a nose down pitching motion that generates negative lift on the blade at this location, which is the result indicated by the flight test data. Near the root of the blade, the addition of elastic motion is not as significant, yielding similar motion as the rigid case. Thus, the addition of elastic motion has the greatest impact at the tip of the blades.

When elastic motion is used, the accuracy of the methods increases. Figures 116, 117, and 118 compare the normal force and the pitching moment results of the free wake and the CFD coupled methods in both polar plot and line plot formats. The free wake coupling yields results that are overall less accurate than the results of the CFD case. The negative lift on the tip of the blade is predicted, as is the peak in the pitching moment at $\psi = 180^o$ at

the tip of the blade. However, the phase of these features lags the flight test data whereas the CFD coupled cases have improved phase prediction as well as improved prediction of the vibratory loading on the advancing blade. The pitching moment results for the CFD coupling are by far better than the prediction using the free wake coupling. In general, using higher fidelity aerodynamic computations corrects the phase lag predicted by the simplified aerodynamic model and also increases the magnitude of the solution closer to the experimentally measured results. Thus, the addition of elastic motion is essential in predicting the main characteristics of the airloads, but the high fidelity of the CFD aerodynamic computations is necessary to predict the magnitude and phase of these features correctly. Although the CFD coupled prediction is better overall, the trimmed hub moments that are provided in Table 7 show that the CFD prediction is farther from the target loads. The free wake coupling predicts hub loads that are closer to the target loads. This has more to do with the interaction of the aerodynamic models with the trimmer than with the fidelity of the aerodynamic predictions. The free wake model uses only the internal airloads model, and is thus more closely coupled with the trimmer, while the CFD model is coupled to the trimmer through the same simplified aerodynamics model using the delta loads coupling method.

The structural moments predicted by the CSD code are also directly affected by the choice of aerodynamic model. Figures 119, 120, and 121 compare the structural bending and torsion moments of the free wake and the CFD coupled methods. The chord bending moment prediction for the CFD airloads has better frequency prediction than the simplified airloads, which have double the frequency. The magnitude prediction is closer to the flight test data near the root of the blade using the CFD loads, but on the outboard section of the blade the magnitude prediction is greater than the flight test data where the simplified airloads are closer. Physically, this moment should become zero at the tip of the blade since this end is free. The flight test data indicate that this is beginning to occur by 80% radial station, but the CFD prediction has not yet begun to diminish by this blade station. The

simplified airload data is of lower magnitude. However, the magnitude of this moment is consistently low at the inboard stations as well. Both airload models have better predictions near the root of the blade where the vibratory loading is not as severe. The flap bending moment predictions for both the CFD and the simplified airloads have approximately the same magnitude. However, there is an improvement in the phase prediction using the CFD airloads. The location of the peak flap bending moment corresponds to the location of the predicted peak normal forces by each of the methods. Neither prediction of the torsional moment has a clear benefit over the other. The CFD prediction has a high magnitude fluctuation on the advancing blade that corresponds to the vibratory nature of the airload profiles. However, the overall magnitude is better predicted by the simplified airloads even though the phase prediction is better with the CFD prediction. Overall, The differences seen between the structural moment predictions correspond with many of the differences seen in the airload predictions. The CFD data has better phase predictions over the free wake coupling, and the free wake coupling does not suffer from the excessive high fluctuations on the advancing blade, though it tends to under predict this region.

Table 7: UH60A case 8424 comparison of trimmed hub loads for different methods

Method	Thrust [lbs]	Roll Moment [ft lbs]	Pitch Moment [ft lbs]
CSD-FW	15967	4097	-3434
CFD-CSD	16077	3790	-1453
Flight Test	15962	4017	-3442

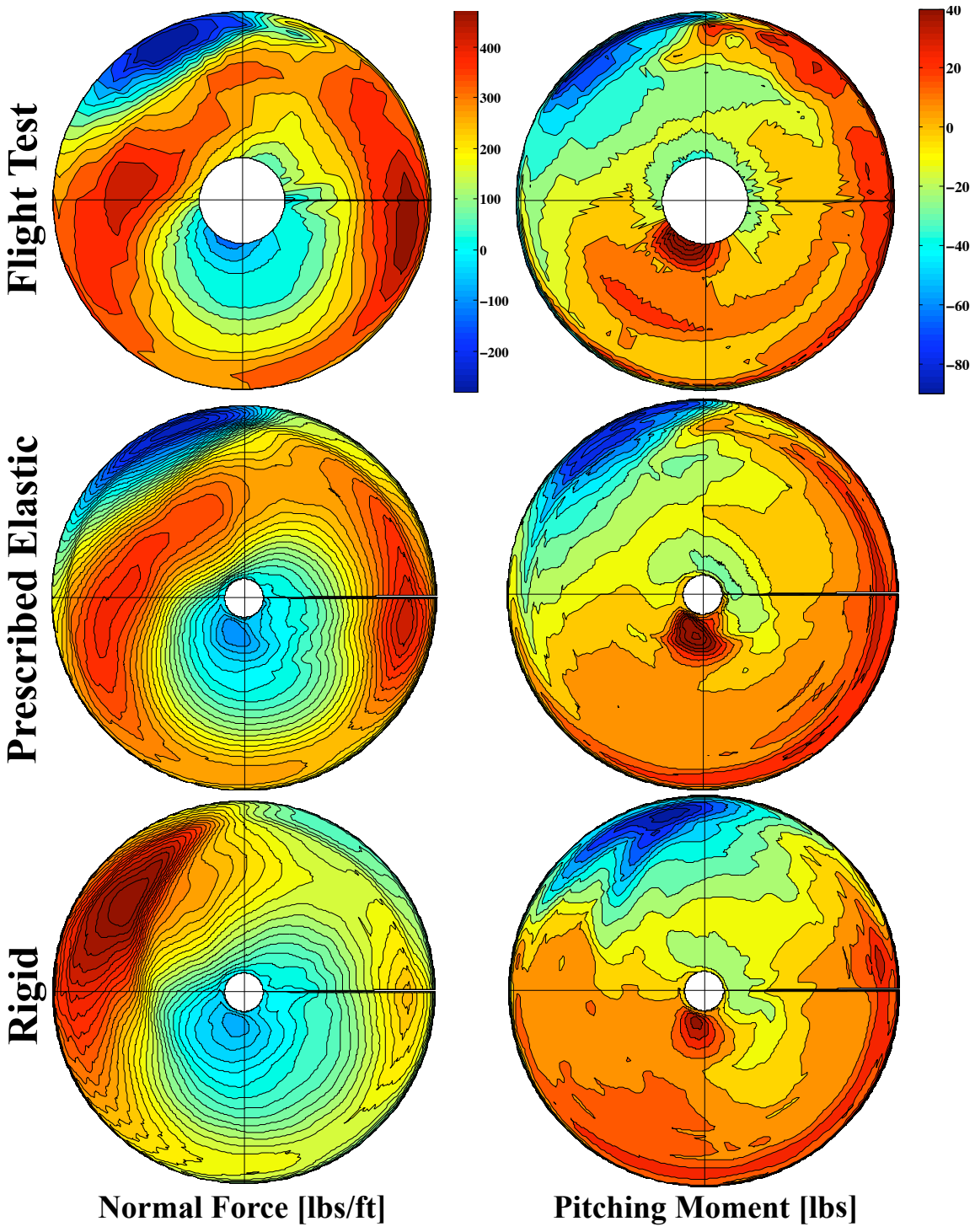


Figure 114: UH60A case 8534 polar plots of the normal force [lbs/ft] and the mean remove pitching moment [lbs] comparing rigid and elastic blade motion, freestream flow from left to right

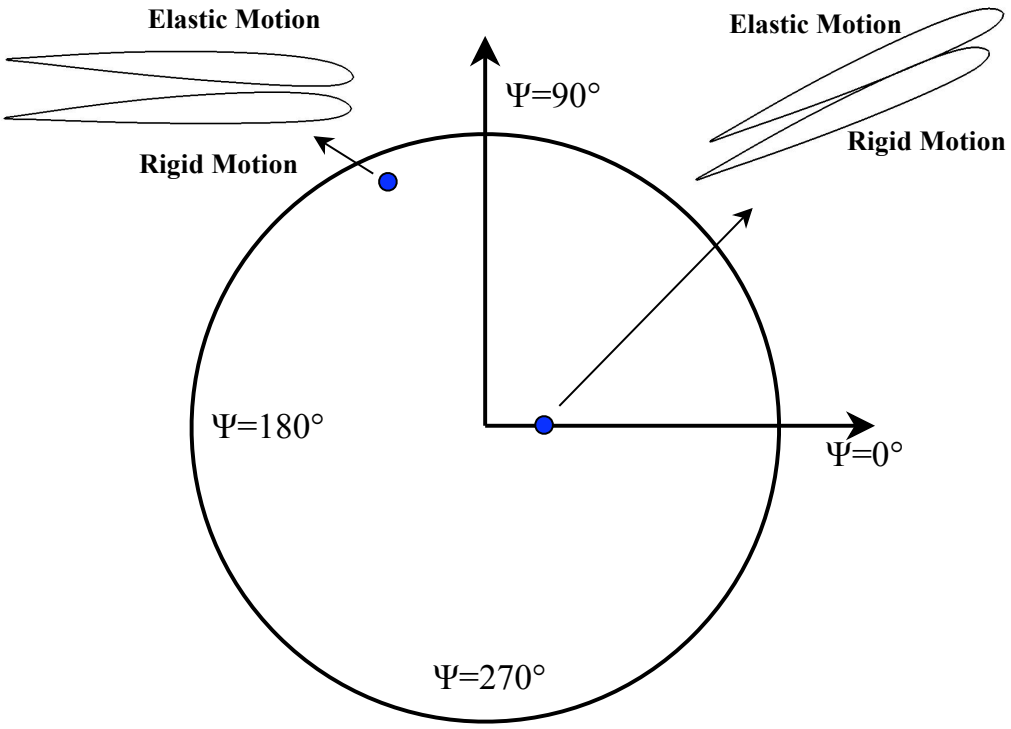


Figure 115: UH60A case 8534 comparison of rigid blade deformation with prescribed elastic blade deformation for selected azimuth and radial positions

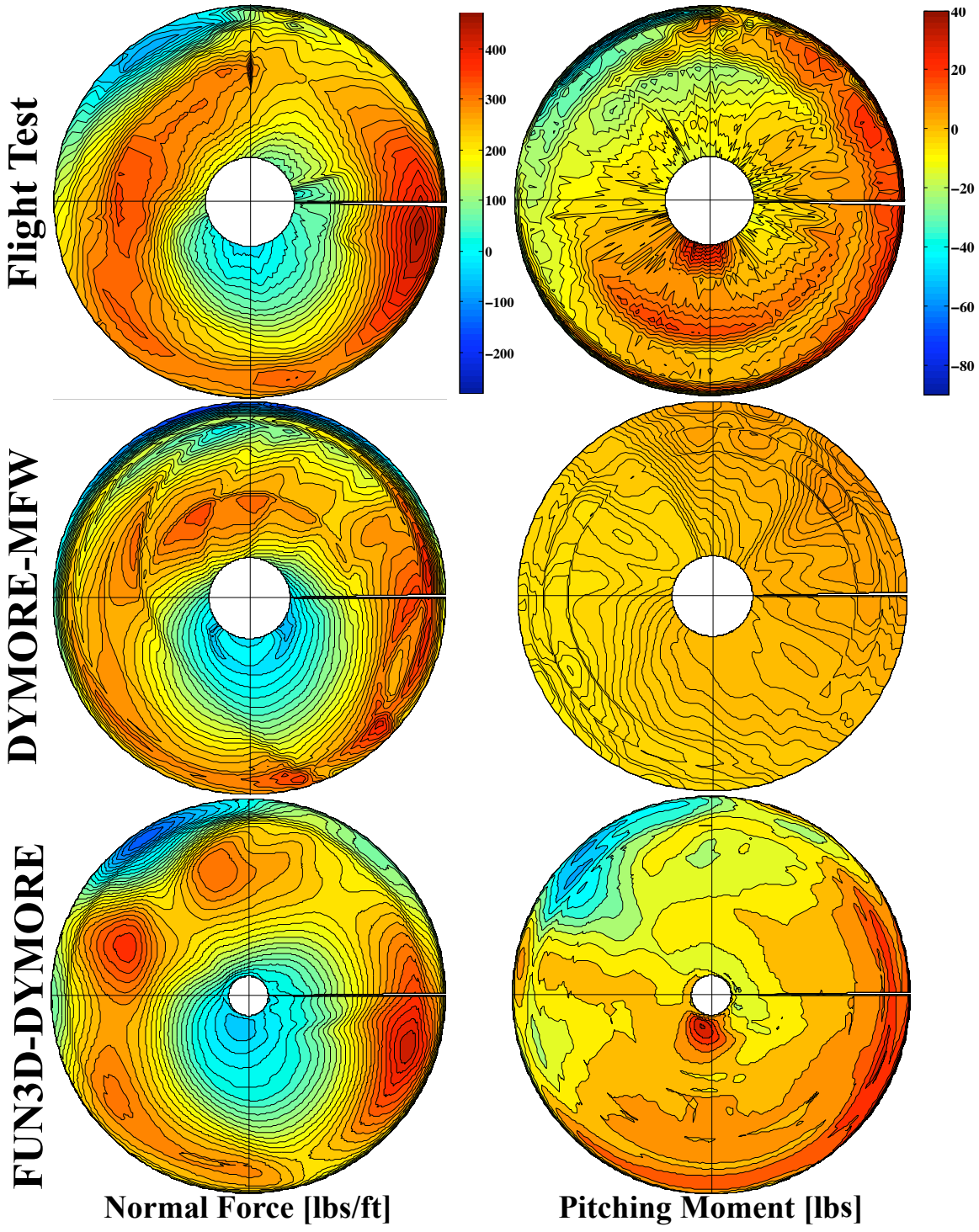


Figure 116: UH60A case 8424 polar plots of the normal force [lbs/ft] and the mean remove pitching moment [lbs] comparing CFD and free wake coupling, freestream flow from left to right

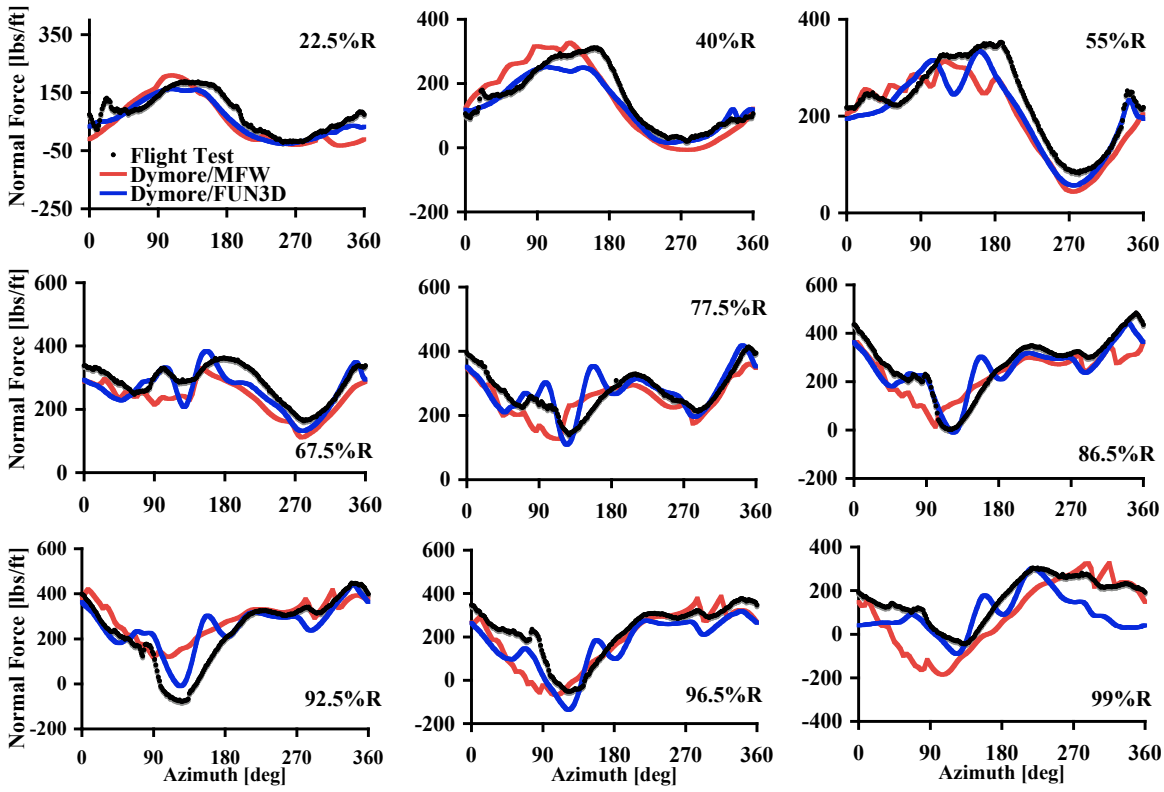


Figure 117: UH60A case 8424 comparison of normal force [lbs/ft] computations using CFD-CSD coupling and CSD-FW coupling

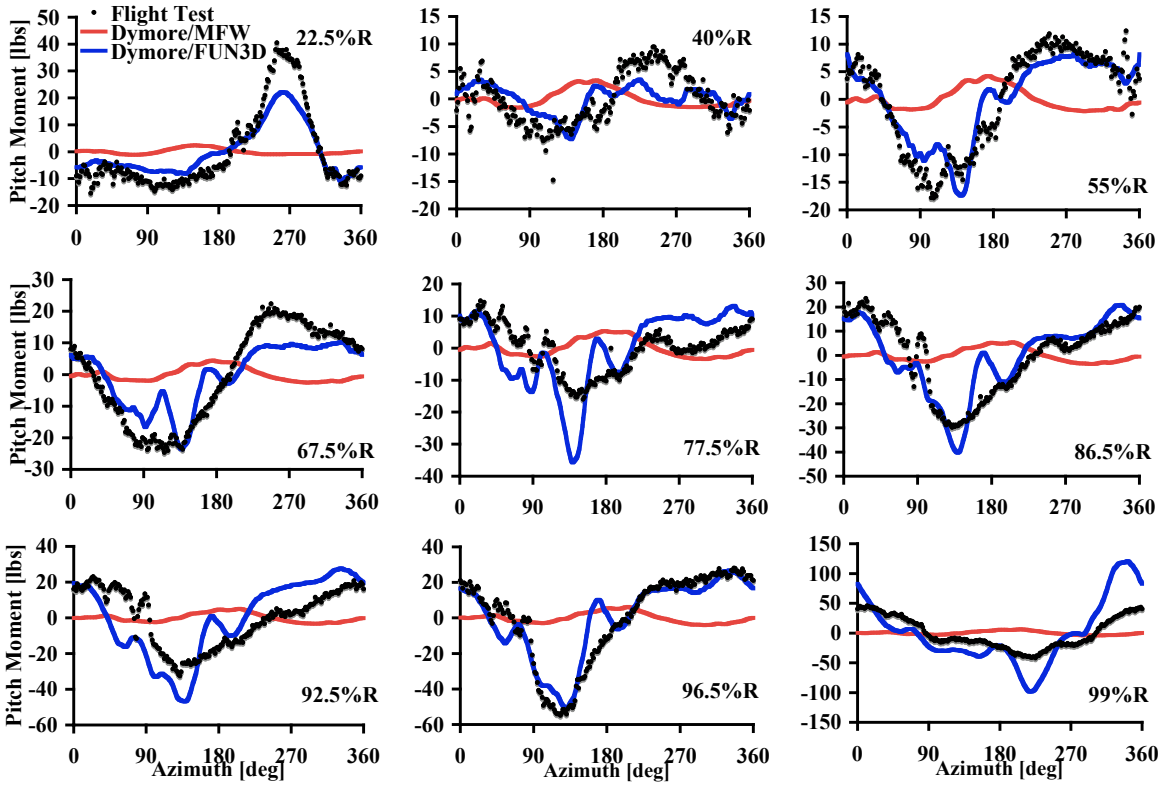


Figure 118: UH60A case 8424 comparison of mean removed pitching moment [lbs] computations using CFD-CSD coupling and CSD-FW coupling

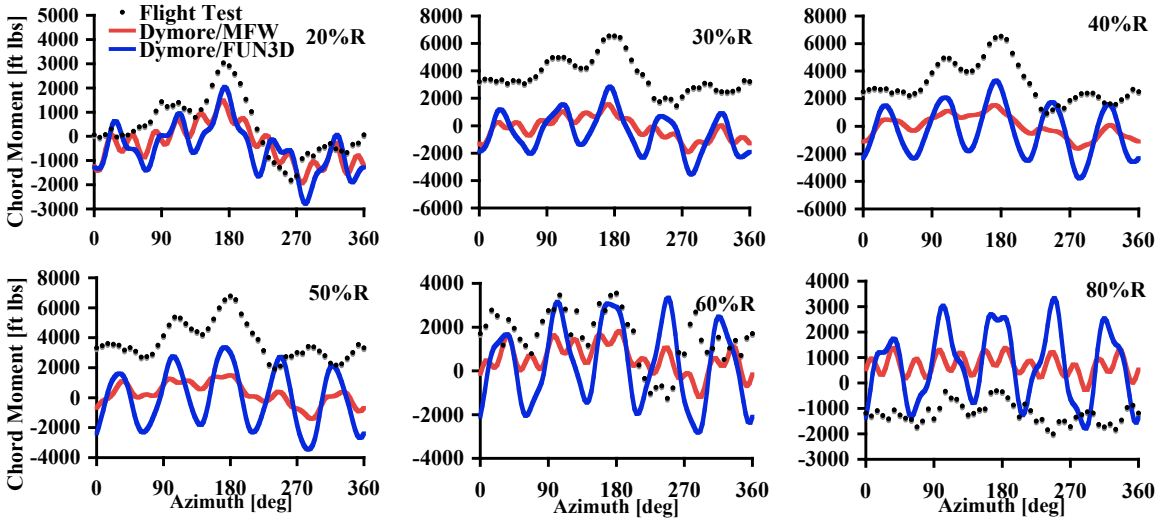


Figure 119: UH60A case 8424 comparison of chord bending moment [ft lbs] computations using CFD-CSD coupling and CSD-FW coupling

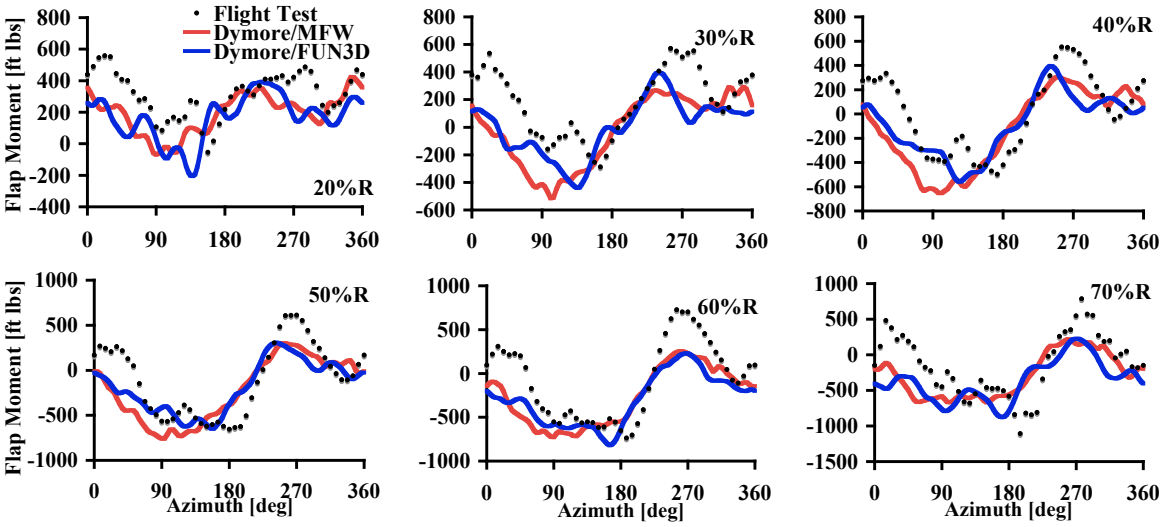


Figure 120: UH60A case 8424 comparison of flap bending moment [ft lbs] computations using CFD-CSD coupling and CSD-FW coupling

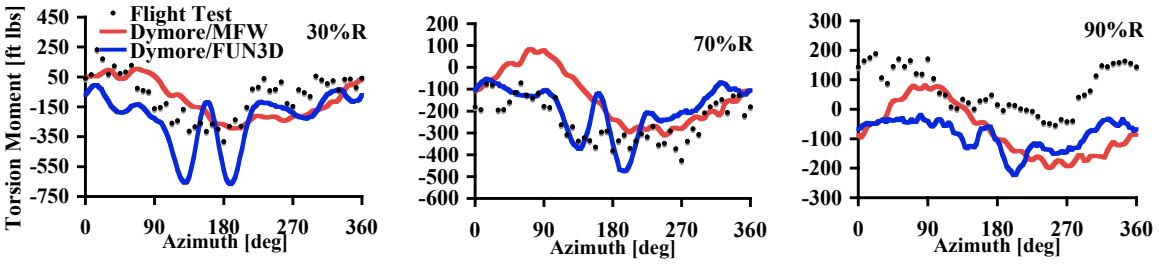


Figure 121: UH60A case 8424 comparison of torsional moment [ft lbs] computations using CFD-CSD coupling and CSD-FW coupling

8.2 Accuracy and Efficiency Analysis

Each of these coupling methods has its own level of accuracy and efficiency, which when compared to one another will yield a qualitative means for determining when it is appropriate to use each method. The procedure used to quantify the accuracy of the coupling algorithms uses a direct comparison of the normal force and pitching moment results. The computed force and moment of each coupling scheme is plotted against the corresponding force and moment from the flight test data in a method developed in reference [89]. Any computed case where the data are identical to the flight test data will fall on the 45^o line. Any deviation from flight test will be indicated by a deviation of the data from this line. The spread of the data away from the line fit is indicated by the R^2 value, and the deviation from the ideal 45^o line is indicated by the slope of the line fit. For the CSD-FW 8534 case in Figure 122 the data has a large spread from the 45^o line indicating that the phase of the solution is different from the flight test data. The pitching moment results have a small spread but a large shift in the slope about zero. These results indicate that the features are captured approximately correctly, but the magnitude of the results is too low. Similar results are obtained for the 9017 case in Figure 123. However, the inability of the simplified aerodynamics to compute dynamic stall has a large impact on the accuracy of the solution. The results for CSD-FW case 8424 in Figure 124 are similar to the 8534 case, but the decrease in the magnitude of the vibratory loading makes this case easier to predict and thus reduces the spread of the data about the 45^o line. The comparison of the rigid motion case in Figure 125 indicates a small spread in the normal force results about the 45^o line except for a small amount of data which is over-predicted. This small amount of data corresponds to the tip of the blade at about $\psi = 135^o$ where the normal force is over-predicted. The pitching moment results indicate that there is both an issue of magnitude and of phase indicated by the spread and slope deviation. The prescribed elastic results in Figure 126 have a much lower spread for the normal force, indicating a much more accurate solution than the other two methods. The pitching moment results still indicate a high spread beneath the

45^o line. However, this comparison is still an improvement over the rigid results. The data comparison of the CFD-CSD coupled case in Figure 127 has the best predictions overall. This conclusion is better shown though the accuracy plot in Figure 128.

The R^2 and the slope parameters are used to compare how close each of the methods is to the experimental data in Figure 128. The ideal result has a slope of one and an R^2 value of one. This point is indicated by the flight test filled circle in Figure 128. The open circles are for the pitching moment comparisons, and the crosses are for the normal force comparisons, where each coupling method is identified by its own color. The fully coupled CFD-CSD method normal force and pitching moment perform the best since they are the shortest distance from the flight test when compared to the other methods. The free wake coupling for 8424 is the next accurate in terms of the normal force, but the low magnitude of the pitching moment data makes the moment calculations the least accurate. The prescribed motion is the next best. However, without prior knowledge of the trimmed elastic motion of the blade, there are few opportunities to use this method in practice. Neglecting the elastic deformation yields the least accurate normal forces, but this method is redeemed by the fact that the higher fidelity CFD computations still compute the magnitude of the pitching moment closer to the test data than the simplified aerodynamic model. The free wake coupled methods closer to the flight envelop boundaries do not perform as well as the case closer to the center of this envelope. This is because the simplified aerodynamic model is not designed to model transonic flow or dynamic stall and thus works better in the region where neither of these phenomena occur.

The distance of each of these points from the ideal will be averaged for each method and plotted in Figure 128. These points are indicated as relative because they are all normalized by the maximum distance and computation time. The final comparison in Figure 129 indicates that the free wake coupling case is the best if efficiency is the most important parameter. The aerodynamic computations provide the most accuracy for the time used for computation. However, if higher accuracy is needed, the prescribed elastic case is the next

best choice, but as stated earlier, without prior knowledge of the trimmed elastic motion of the blade there are few opportunities to use this method in practice, if even higher accuracy is needed than the full CFD-CSD coupling should be used. This method is the least efficient, but it has the most accurate predictions. The rigid motion, although it provides better efficiency results than prescribed or computed elastic motion, should not be used unless elastic deformation is not a significant component of the blade motion.

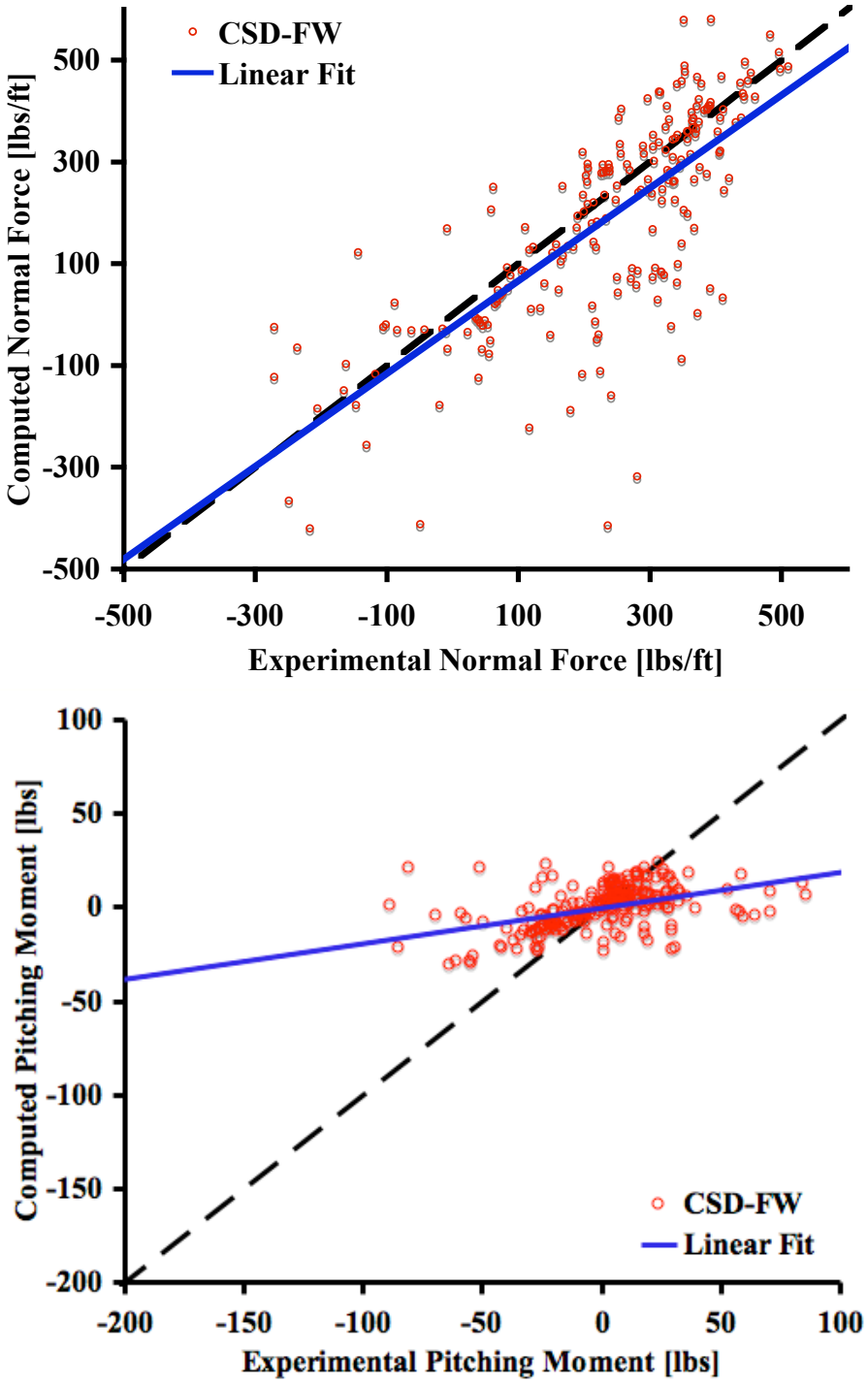


Figure 122: UH60A case 8534 normal force [lbs/ft] and mean removed pitching moment [lbs] comparison of CSD-FW with flight test

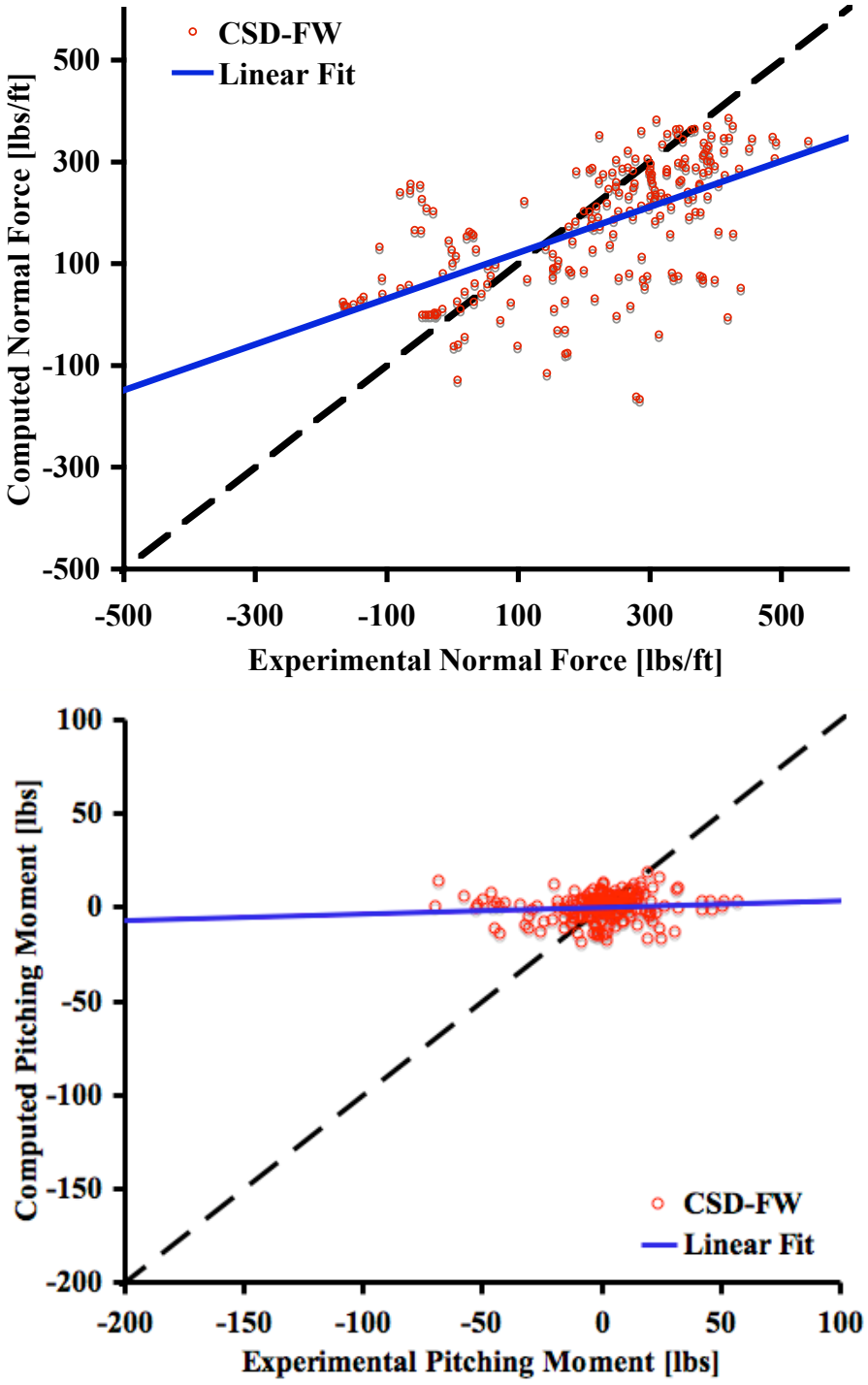


Figure 123: UH60A case 9017 normal force [lbs/ft] and mean removed pitching moment [lbs] comparison of CSD-FW with flight test

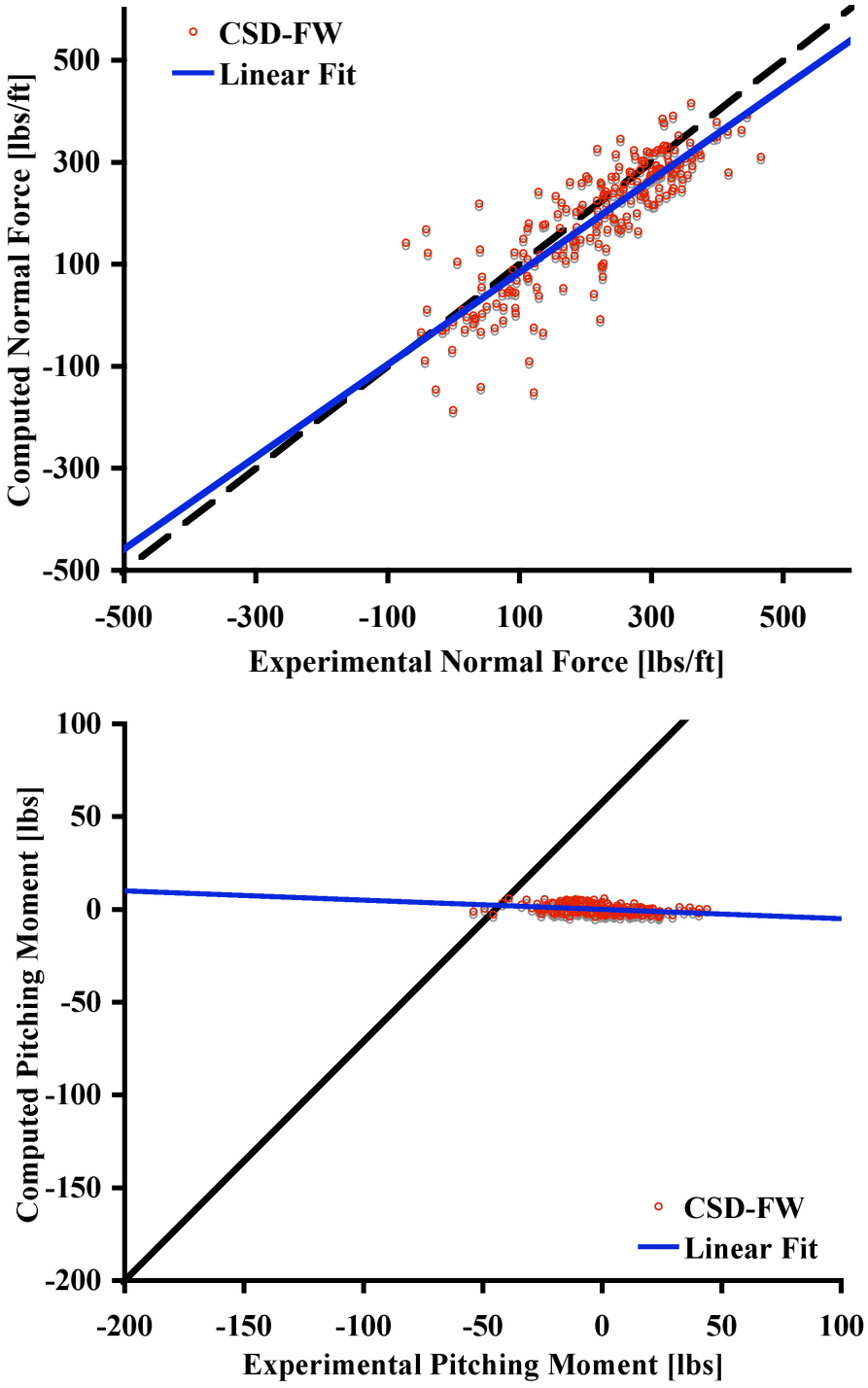


Figure 124: UH60A case 8424 normal force [lbs/ft] and mean removed pitching moment [lbs] comparison of CSD-FW with flight test

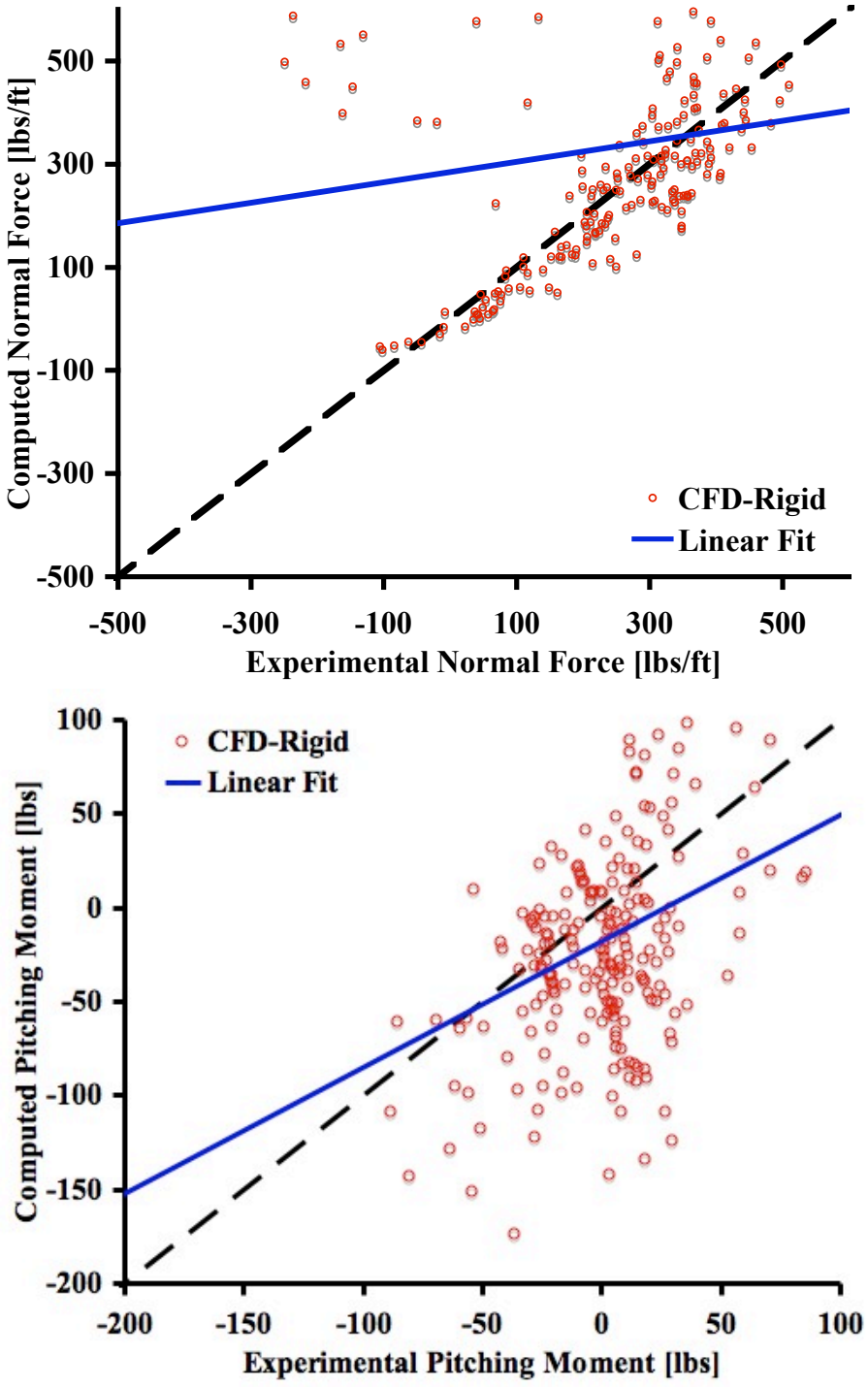


Figure 125: UH60A case 8534 normal force [lbs/ft] and mean removed pitching moment [lbs] comparison of CFD-Rigid motion with flight test

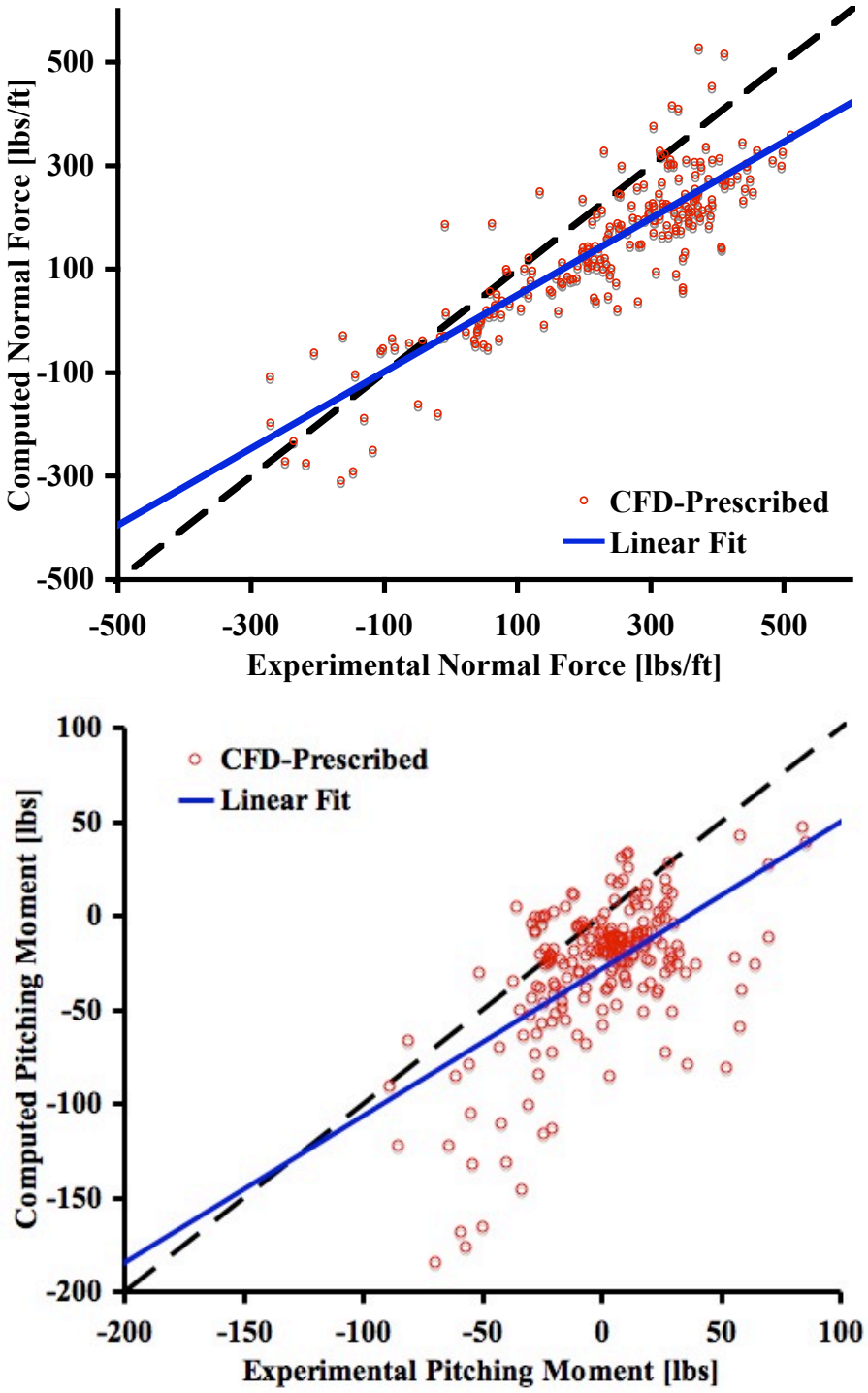


Figure 126: UH60A case 8534 normal force [lbs/ft] and mean removed pitching moment [lbs] comparison of CFD-Prescribed motion with flight test

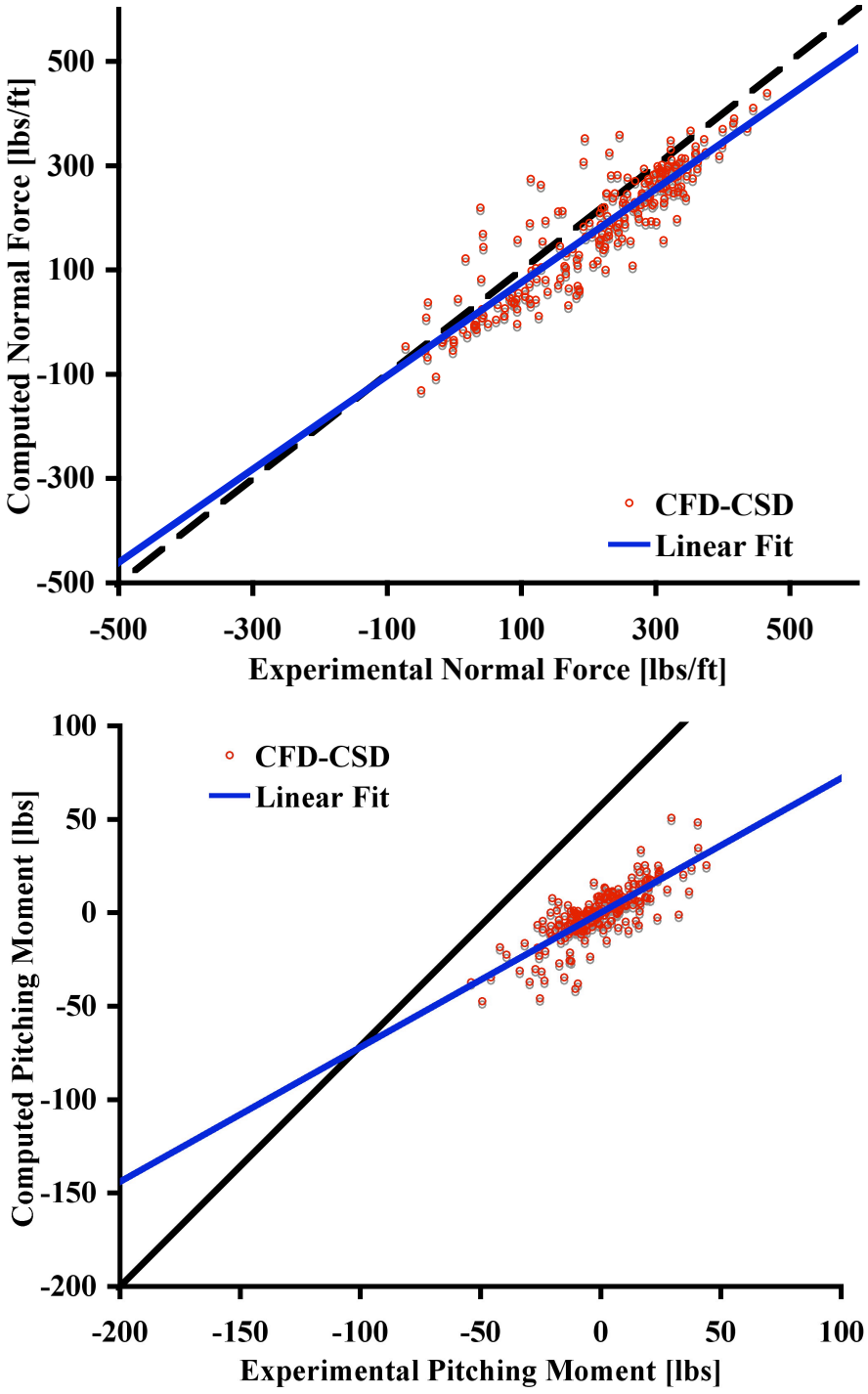


Figure 127: UH60A case 8424 normal force [lbs/ft] and mean removed pitching moment [lbs] comparison of CFD-CSD coupling with flight test

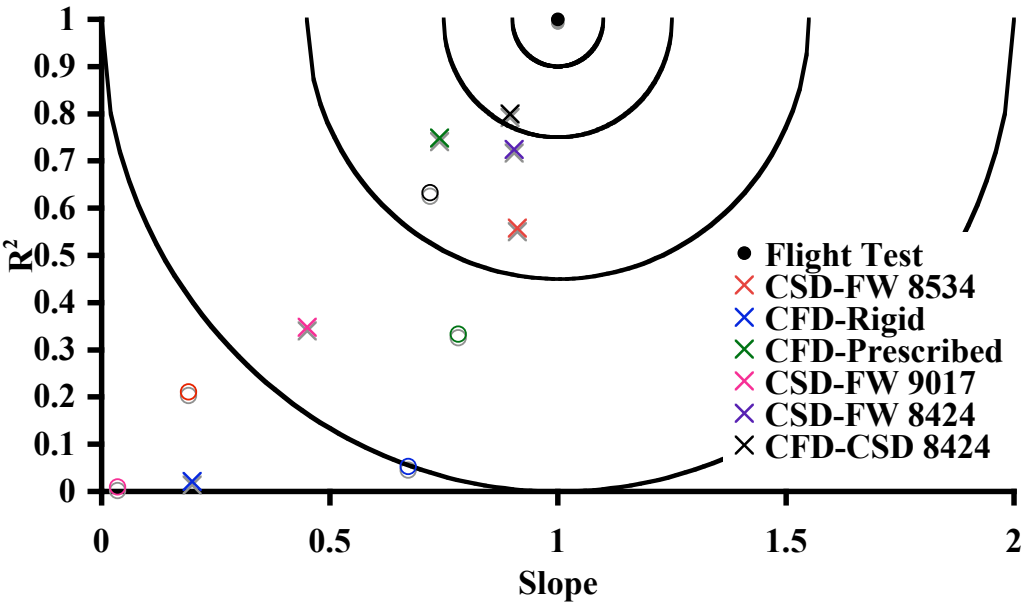


Figure 128: UH60A comparison of spread of all computed cases with flight test, where the X's are the normal force statistics and the O's are the pitching moment statistics

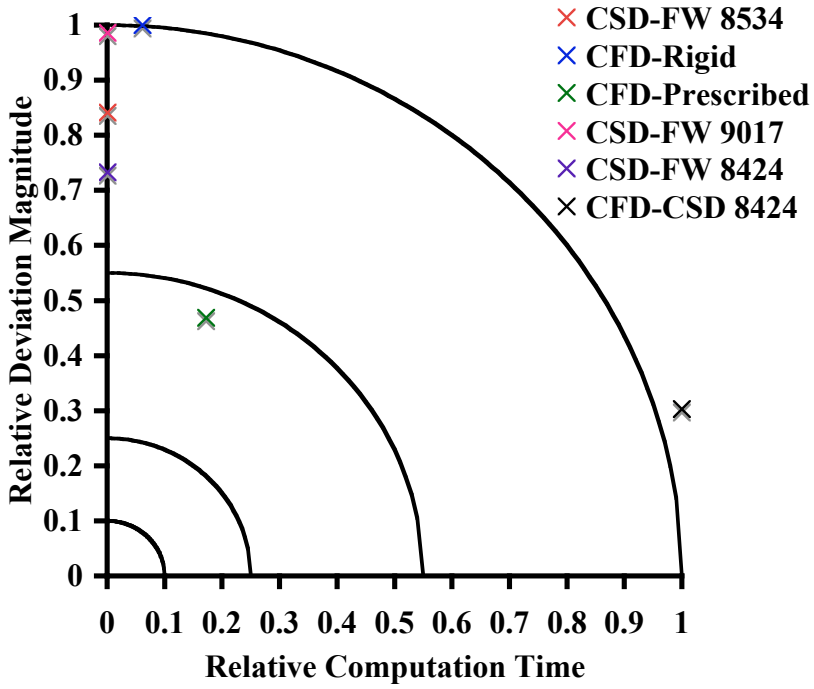


Figure 129: UH60A comparison of efficiency versus accuracy for a variety of coupling methods

CHAPTER 9

CONCLUSIONS

As the tools available to the rotorcraft community grow in number and become more sophisticated, more accurate and efficient rotor airload predictions will be obtained. There are many fronts from which this problem may be approached, including individual improvements to CFD and CSD codes. Improvements to the CFD code through introduction of time-accurate low Mach number preconditioning, application of steady-state grid adaptation, and steady-state low Mach number preconditioning have been shown. The preconditioning methods reduced the numerical fluctuations associated with using the compressible equations in low speed flow, and the development of the time-accurate LMP method extends its use to rotating cases, which is an important aspect of rotor analysis. The grid adaptation effectively refines the wake, picking up features of the flow that the unadapted grid misses. However, this method is currently only suitable for steady-state cases as it has not been implemented to account for moving grids. In general, all of these methods are shown to improve the CFD predictions, however, the state of the art of these rotor airload predictions is to couple aerodynamic and structural computations into the same simulation. This research presents a range of these methods from the most approximate CSD-FW coupling, to the most complex CFD-CSD coupling. Depending on the accuracy needs of the user, being it a quick initial computation from a comprehensive analysis, or a detailed analysis, a method is available that best suits the situation. There are a number of conclusions that may be drawn from this research

- For CSD-FW coupling the angle of attack method is the recommended algorithm since it yields the best results.
- Overall, if pitching moment predictions are needed than the FW coupled method

should not be used as its predictions are not anywhere close to the flight test data.

• When looking at the independent FW and CSD computations, the coupled method performs better than the isolated component computations.

• When setting up the FW model, one wake trailer is sufficient to model the flow. However, the best model is obtained by adding wake trailers at each location where there is a geometry change.

• The CSD-FW computations are more appropriate for cases where there are not significant assumptions made in the 2D aerodynamic model, such as dynamic stall.

• LMP is able to preserve the rotor wake vortices as they propagate in a low speed flow better than the un-preconditioned compressible equations in regimes where there are both incompressible and compressible Mach numbers.

• The steady grid adaptation scheme is able to preserve the time averaged wake geometry longer than the unadapted method as the wake propagates away from the rotor.

• Using intermediate methods such as CFD-Rigid computations are not always suitable if the elastic deformations are a significant component of the motion. If the elastic deformations are significant, than important phenomena may be missed if the correct structural deflections are not used. However, if the blades are approximately rigid than this method may be used with more confidence.

• The prescribed elastic deformation method has improved efficiency over the CSD coupled method. However, the use of this method requires the availability of blade deformation data.

• The CFD-CSD computations are the most accurate for elastic cases, because this method makes the fewest assumptions, but the time to obtain the results is the longest of all methods studied.

• When comparing the use of structured and unstructured CFD methodologies, the unstructured method has improved prediction of the structural response. This is likely because of the absence of additional dissipation terms in the CFD solver.

9.1 Recommendations for Future Work

• DYMORE Simplified Aerodynamic Models

In order to reduce the number of uncertainties when comparing the CSD-FW results to the experimental data, the known assumptions that reduce the accuracy of the situation must be accounted for. The need to enhance the unsteady non-linear (high angle of attack) computations, so that the reverse flow region and dynamic stall phenomena can be modeled, will improve the quality of the predictions.

• Alternative Wake Methods

Free wake methods are not the only wake models available. The state of the art of wake modeling has recently expanded to include vortex particle and vorticity transport methods. The coupling of the comprehensive code with a more advanced Navier-Stokes based wake model is a topic of future work.

• FUN3D Overset Computations

Currently the overset computations in FUN3D are more time consuming than the flow solver because a DCI file must be computed for each deformed configuration in the simulation. For a large grid this can add an additional 5-8 minutes to each time step from a whole revolution. These computations can be made more efficient by implementing a parallel version of the hole cutting software that has been converted into a subroutine callable by the CFD code.

• CFD-CSD Maneuvering Cases

Future work to implement a maneuvering capability into the CFD-CSD coupling can be achieved by implementing tight coupling. This capability is not present at this time, but would enhance the prediction capabilities of the unstructured CFD-CSD coupling.

• CFD Dissipation and Damping Study

A study of the impact of damping on the FUN3D simulation and further studies into the impact of numerical dissipation on the CFD structured methods would be useful in developing improved CSD coupled methods in the future.

• FUN3D Unsteady Grid Adaptation

The ability of the feature-based grid adaptation algorithm in FUN3D has been proven to work well for steady-state cases. However, to apply this ability to a moving rotor case needs more work. The grid adaptation software would need to be developed such that it could run concurrently with the flow solver. This would allow the grid to be adapted as the blade moved without having to stop the simulation at every time step. The hole cutting algorithm would also need to be implemented internally so that the DCI files could be computed as the grids are changed, without having to write out new component grids to get a new DCI file at each time step. Since there is no grid refinement in the viscous boundary layer or on the fuselage surface, the full impact of refining the wake is not seen in the surface results. There are limited improvements in this viscous region. Including the ability to refine the viscous wake would further improve the surface predictions.

• FUN3D Mixed Mach Capability

The ability to perform mixed Mach computations instead of LMP would have the advantage of eliminating the empirical β parameter. This parameter is automatically set to one in regions of higher Mach numbers, effectively turning off preconditioning in these areas; however, the use of the preconditioning matrix in general introduces numerical error into the governing equations. For high Mach cases in the transonic regime the range of Mach numbers is much larger since the Mach numbers would need to cover the range between about 0.3 (the end of incompressible flow) to about 0.85 (the start of transonic flow) to be useful. This range of $\beta's$ will create a more unbalanced preconditioning matrix resulting in more numerical error. The mixed Mach method uses the un-preconditioned governing equations where appropriate, eliminating any uncertainty with regard to shock wave modeling since the un-preconditioned compressible equations are used in these regions. Thus, the impact of using LMP on high speed flows with shocks needs to be further evaluated to test the impact of the numerical error on these cases.

REFERENCES

[1] Biedron, R., Vatsa, V., and Atkins, H., "Simulation of Unsteady Flows Using an Unstructured Navier-Stokes Solver on Moving and Stationary Grids," in *23rd AIAA Applied Aerodynamics Conference*, (Toronto, Canada), American Institute of Aeronautics and Astronautics, June 2005.

[2] O'Brien, D., *Analysis of Computational Modelling Techniques for Complete Rotorcraft Configurations*. PhD thesis, Georgia Institute of Technology, 2006.

[3] Eric Lynch Internal Communication, May 2007.

[4] Shelton, A., Abras, J., Hathaway, B., Sanchez-Rocha, M., Smith, M., and Menon., S., "An Investigation of the Numerical Prediction of Static and Dynamic Stall," in *Proceedings of the 61st Annual Forum*, (Grapevine, TX), American Helicopter Society, June 2005.

[5] Kunz, D. L., "Comprehensive Rotorcraft Analysis: Past, Present and Future," in *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, (Austin, TX), April 2005.

[6] Bir, G. and Chopra, I., "Development of UMARC (University of Maryland Advanced Rotorcraft Code)," in *Proceedings of the 46th Annual Forum*, (Washington D.C.), American Helicopter Society, May 1990.

[7] Johnson, W., "Technology Drivers in the Development of CAMRAD II," in *Aeromechanics Specialists Conference*, (San Francisco, CA), American Helicopter Society, January 1994.

[8] Johnson, W., "Rotorcraft Dynamics Models for a Comprehensive Analysis," in *Proceedings of the 54th Annual Forum*, (Washington D.C.), American Helicopter Society, May 1998.

[9] Sopher, R. and Hallock, D., "Time-History Analysis for Rotorcraft Dynamics Based on a Component Approach," in *2nd Decennial Specialists' Meeting on Rotorcraft Dynamics*, (Moffett Field, CA), November 1984.

[10] Benoit, B., Dequin, A. M., Kampa, K., Grunhagen, W., Basset, P. M., and Gimonet, B., "HOST: A General Helicopter Simulation Tool For Germany and France," in *Proceedings of the 56th Annual Forum*, (Virginia Beach, VA), American Helicopter Society, May 2000.

[11] Bauchau, O. A., Bottasso, C. L., and Nikishkov, Y. G., "Modeling Rotorcraft Dynamics with Finite Element Multibody Procedures," *Mathematical and Computer Modeling*, vol. 33, pp. 1113–1137, 2001.

[12] Jespersen, D., Pulliam, T., and Buning, P., "Recent enhancements to OVERFLOW," in *35th Aerospace Sciences Meeting and Exhibit*, (Reno, NV), American Institute of Aeronautics and Astronautics, January 1997.

[13] Buning, P., Chiu, I., Obayashi, S., Rizk, Y., and Steger, J., "Numerical Simulation of the Integrated Space Shuttle Vehicle in Ascent," in *AIAA Atmospheric Flight Mechanics Conference*, (Minneapolis,Minnesota), American Institute of Aeronautics and Astronautics, August 1988.

[14] Bonhaus, D. L., "An Upwind Multigrid Method for Solving Viscous Flows on Unstructured Triangular Meshes," Master's thesis, George Washington University, 1993.

[15] Anderson, W. K. and Bonhaus, D. L., "An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids," *Computers and Fluids*, vol. 23, no. 1, pp. 1–21, 1994.

[16] Anderson, W. K., Rausch, R. D., and Bonhaus, D. L., "Implicit/Multigrid Algorithms for Incompressible Turbulent Flows on Unstructured Grids," *Journal of Computational Physics*, vol. 128, no. 2, pp. 391–408, 1996.

[17] "http://fun3d.larc.nasa.gov." Website, Last Accessed, June 2007.

[18] Sitaraman, J., Datta, A., Baeder, J., and Chopra, I., "Coupled CFD/CSD Prediction of Rotor Aerodynamic and Structural Dynamic Loads for Three Critical Flight Conditions," in *31st European Rotorcraft Forum*, (Florence, Italy), September 2005.

[19] Egolf, A., Wake, B., and Berezin, C., "Recent Rotor Wake Simulation and Modeling Studies at United Technologies Corporation(Invited)," in *38th Aerospace Sciences Meeting and Exhibit*, (Reno, NV), American Institute of Aeronautics and Astronautics, January 2000.

[20] Datta, A., Sitaraman, J., Baeder, J. D., and Chopra, I., "Analysis Refinements for Prediction of Rotor Vibratory Loads in High-Speed Forward Flight," in *Proceedings of the 60th Annual Forum*, (Baltimore, MD), American Helicopter Society, June 2004.

[21] Abras, J., Smith, M., and Bauchau, O., "Rotor Airloads Prediction Using Coupled Finite Element and Free Wake Analysis," in *Proceedings of the 62nd Annual Forum*, (Phoenix, AZ), American Helicopter Society, May 2006.

[22] Johnson, W., "A General Free Wake Geometry Calculation for Wings and Rotors," in *Proceedings of the 51st Annual Forum*, (Fort Worth, TX), American Helicopter Society, May 1995.

[23] Bousman, W. G., Young, C., Gilbert, N., Toulmay, F., Johnson, W., and Riley, M. J., "Correlation of PUMA Airloads - Lifting-Line and Wake Calculation," Tech. Rep. TM-102212, NASA, 1989.

[24] Tung, C., Caradonna, F. X., and Johnson, W., "The Prediction of Transonic Flows on Advancing Rotors," *Journal of the American Helicopter Society*, vol. 31, pp. 4–9, 1986.

[25] Hassan, A., Charles, B., Tadghighi, H., and Burley, C., "Consistent Approach for Modeling the Aerodynamics of Self-Generated Rotor Blade-Vortex Interactions," *Journal of the American Helicopter Society*, vol. 41, no. 2, pp. 74–84, 1996.

[26] Strawn, R. C., Desopper, A., Miller, J., and Jones, A., "Correlation of PUMA Airloads - Evaluation of CFD Prediction Methods," Tech. Rep. TM-102226, NASA, 1989.

[27] Torok, M. S. and Berezin, C. R., "Aerodynamic and Wake Methodology Evaluation Using Model UH60A Experimental Data," in *Proceedings of the 48th Annual Forum*, (Washington, D.C), American Helicopter Society, June 1992.

[28] Lee, C. S., Saberi, H., and Ormiston, R. A., "Aerodynamic and Numerical Issues for Coupling CFD into Comprehensive Rotorcraft Analysis," in *Proceedings of the 53rd Annual Forum*, (Virginia Beach, VA), American Helicopter Society, April 1997.

[29] Manke, J. W., Hirsh, J. E., Oh, B. K., Wicks, T. M., and Dadone, L., "Improved Rotor Tip-Relief Modeling by Coupling Comprehensive Rotor Analysis and Rotor Aerodynamics Codes," *Advances in Engineering Software*, vol. 29, pp. 475–480, 1998.

[30] Alonso, J. J., Sheffer, S. G., Martinelli, L., and Jameson, A., "Parallel Unsteady Simulation of the Flow Through a Helicopter Rotor in Hover Including Aeroelastic Effects," in *1st AFOSR Conference on Dynamic Motion CFD*, (New Brunswick), Rutgers University, June 1996.

[31] Smith, M. J., *A Fourth-Order Euler/Navier-Stokes Prediction Method for the Aerodynamics and Aerolasticity of Hovering Rotor Blades*. PhD thesis, Georgia Institute of Technology, 1994.

[32] Bauchau, O. A. and Ahmad, J. U., "Advanced CFD and CSD Methods for Multidisciplinary Applications of Rotorcraft Problems," in *6th Symposium on Multidisciplinary Analysis and Optimization*, (Seattle, WA), American Institute of Aeronautics and Astronautics, September 1996.

[33] Murty, H., Bottasso, C., Dindar, M., Shephard, M., and Bauchau, O., "Aeroelastic Analysis of Rotor Blades Using Nonlinear Fluid/Structure Coupling," in *Proceedings of the 53rd Annual Forum*, (Virginia Beach, VA), American Helicopter Society, April 1997.

[34] Servera, G., Beaumier, P., and Costes, M., "A Weak Coupling Method Between the Dynamics Code HOST and the 3D Unsteady Euler Code WAVES," *Aerospace Science and Technology*, vol. 5, pp. 397–408, 2001.

[35] Beaumier, P., Costes, M., Rodriguez, B., Poinot, M., and Cantaloube, B., "Weak and Strong Coupling Between the elsA CFD Solver and the Host Helicopter Comprehensive Code," in *31st European Rotorcraft Forum*, (Florence, Italy), September 2005.

[36] Potsdam, M., Yeo, H., and Johnson, W., "Rotor Airloads Prediction Using Loose Aerodynamic/Structural Coupling," in *Proceedings of the 60th Annual Forum*, (Baltimore, MD), American Helicopter Society, June 2004.

[37] Datta, A. and Chopra, I., "Prediction of UH-60A Dynamic Stall Loads in High Altitude Level Flight Using CFD/CSD Coupling," in *Proceedings of the 61st Annual Forum*, (Grapevine, TX), American Helicopter Society, June 2005.

[38] Bhagwat, M., Dimanlig, A., Saberi, H., Meadowcroft, E., Panda, B., and Strawn, R., "CFD/CSD Coupled Trim Solution for the Dual-Rotor CH-47 Helicopter Including Fuselage Modeling," in *AHS Specialist's Conference on Aeromechanics*, (San Francisco, CA), American Helicopter Society, January 2008.

[39] Hill, M. J., *Coupled Fluid - Structure Simulations of Helicopter Rotors*. PhD thesis, Pennsylvania State University, 2006.

[40] Abras, J., Lynch, E., and Smith, M., "Advances in Rotorcraft Simulations with Unstructured CFD," in *Proceedings of the 63rd Annual Forum*, (Virginia Beach, VA), American Helicopter Society, May 2007.

[41] Datta, A., Nixon, M., and Chopra, I., "Review of Rotor Loads Prediction with the Emergence of Rotorcraft CFD," in *31st European Rotorcraft Forum*, (Florence, Italy), September 2005.

[42] Nygaard, T. A., Saberi, H., Ormiston, R., Strawn, R. C., and Potsdam, M., "CFD and CSD Coupling Algorithms and Fluid Structure Interface for Rotorcraft Aeromechanics in Steady and Transient Flight Conditions," in *Proceedings of the 62nd Annual Forum*, (Phoenix, AZ), American Helicopter Society, May 2006.

[43] Bauchau, O. A., "Computational Schemes for Flexible, Nonlinear Multi-Body Systems," *Multibody System Dynamics*, vol. 2, pp. 169–225, 1998.

[44] Berdichevsky, V. L., "On the Energy of an Elastic Rod," *Journal of Applied Mathematics and Mechanics*, vol. 45, pp. 518–529, 1982.

[45] Cesnik, C. E. S. and Hodges, D. H., "VABS: A New Concept for Composite Rotor Blade Cross-Sectional Modeling," *Journal of the American Helicopter Society*, vol. 42, no. 1, pp. 27–38, 1997.

[46] Bauchau, O. A., Choi, J. Y., and Bottasso, C. L., "On the Modeling of Shells in Multibody Dynamics," *Multibody System Dynamics*, vol. 8, no. 4, pp. 459–489, 2002.

[47] Peters, D. A., Barwey, D., and Johnson, M. J., "Finite-State Airloads Modeling with Compressibility and Unsteady Free-Stream," in *6th International Workshop on Dynamics and Aeroelastic Stability Modeling of Rotorcraft Systems*, (Los Angeles, CA), American Helicopter Society, November 1995.

[48] Leishman, J. G. and Beddoes, T. S., "A Generalized Model for Airfoil Unsteady Aerodynamic Behavior and Dynamic Stall Using the Indicial Method," in *Proceedings of the 42nd Annual Forum*, (Washington D.C), American Helicopter Society, June 1986.

[49] Petot, D., "Differential Equation Modeling of Dynamic Stall," *Recherche Aerospatiale*, vol. 5, pp. 59–72, 1989.

[50] Leishman, J. G. and Beddoes, T. S., "A Semi-Empirical Model for Dynamic Stall," *Journal of the American Helicopter Society*, vol. 4, pp. 3–17, 1989.

[51] Peters, D. A., Hsieh, M. A., and Torrero, A., "A State-Space Airloads Theory for Flexible Airfoils," in *Proceedings of the 62th Annual Forum*, (Phoenix, AZ), American Helicopter Society, May 2006.

[52] Peters, D. and He, C., "Finite State Induced Flow Models Part II: Three-Dimensional Rotor Disk," *Journal of Aircraft*, vol. 32, no. 2, pp. 323–333, 1995.

[53] Peters, D. and Barwey, D., "A General Theory of Rotorcraft Trim," *Mathematical Problems in Engineering*, vol. 2, pp. 1–34, 1996.

[54] Leishman, J. G., Bhagwat, M. J., and Bagai, A., "Free-Vortex Filament Methods for the Analysis of Helicopter Rotor Wakes," *Journal of Aircraft*, vol. 39, no. 5, pp. 759–775, 2002.

[55] Ramasamy, M. and Leishman, J. G., "Reynolds Number Based Blade Tip Vortex Model," in *Proceedings of the 61st Annual Forum*, (Grapevine, TX), American Helicopter Society, June 2005.

[56] Weissinger, J., "The Lift Distribution of Swept-Back Wings," Tech. Rep. TM 1120, NACA, 1947.

[57] Bagai, A., *Contributions to the Mathematical Modeling of Rotor Flow-Fields Using a Pseudo-Implicit Free-Wake Analysis*. PhD thesis, University of Maryland, 1995.

[58] Bhagwat, M. J., *Mathematical Modeling of the Transient Dynamics of Helicopter Rotor Wakes Using a Time-Accurate Free-Vortex Method*. PhD thesis, University of Maryland, 2001.

[59] Leishman, J. G., "Subsonic Unsteady Aerodynamics Caused by Gusts Using the Indicial Method," *Journal of Aircraft*, vol. 33, no. 5, pp. 869–879, 1996.

[60] Spalart, P. R. and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *La Recherche Aérospatiale*, vol. 1, no. 5, pp. 5–21, 1994.

[61] Menter, F. R., "Influence of Freestream Values on k-omega Turbulence Model Predictions," *AIAA Journal*, vol. 30, no. 6, pp. 1657–1659, 1992.

[62] Noack, R., "DiRTlib: A Library to Add an Overset Capability to Your Flow Solver," in *17th AIAA Computational Fluid Dynamics Conference*, (Toronto, Canada), American Institute of Aeronautics and Astronautics, June 2005.

[63] Noack, R., "SUGGAR: A General Capability for Moving Body Overset Grid Assembly," in *17th AIAA Computational Fluid Dynamics Conference*, (Toronto, Canada), American Institute of Aeronautics and Astronautics, June 2005.

[64] Bibb, K. L., Gnoffo, P. A., Park, M. A., and Jones, W. T., "Parallel, Gradient-Based Anisotropic Mesh Adaptation for Re-entry Vehicle Configurations," in *9th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, (San Francisco, CA), June 2006.

[65] Park, M. A., "Three-Dimensional Turbulent RANS Adjoint-Based Error Correction," in *6th AIAA Computational Fluid Dynamics Conference*, (Orlando, FL), June 2003.

[66] Tung, C., Caradonna, F. X., and Johnson, W. R., "The Prediction of Transonic Flows on an Advancing Rotor," in *Proceedings of the 40th Annual Forum*, (Arlington, VA), American Helicopter Society, May 1984.

[67] Abras, J. and Smith, M., "Rotorcraft Methodology for Unstructured CFD-CSD Coupling," in *AHS Specialist's Conference on Aeromechanics*, (San Francisco, CA), American Helicopter Society, January 2008.

[68] Boschitsch, A. H., Usab, W. J., and Epstein, R. J., "Fast Lifting Panel Method," in *14th Computational Fluid Dynamics Conference*, (Norfolk, VA), American Institute of Aeronautics and Astronautics, June 1999.

[69] Wachspress, D. A., Quackenbush, T. R., and Boschitsch, A. H., "Rotorcraft Interactional Aerodynamics Calculations with Fast Vortex/Fast Panel Methods," in *Proceedings of the 56th Annual Forum*, (Virginia Beach, VA), American Helicopter Society, June 2000.

[70] Johnson, W., *Helicopter Theory*. New York: Dover Publications, Inc., 1994.

[71] Edwards, J. R. and Thomas, J. L., "Development and Investigation of $O(Nn^2)$ Preconditioned Multigrid Solvers for the Euler and Navier-Stokes Equations," in *14th AIAA Computational Fluid Dynamics Conference*, (Norfolk, VA), June 1999.

[72] Withington, J. P., Shuen, J. S., and Yang, V., "A Time Accurate, Implicit Method for Chemically Reacting Flows at All Mach Numbers," in *29th Aerospace Sciences Meeting*, (Reno, NV), American Institute of Aeronautics and Astronautics, January 1991.

[73] Potsdam, M. A., Sankaran, V., and Pandya, S. A., "Unsteady Low Mach Number Preconditioning with Application to Rotorcraft Flows," in *18th AIAA Computational Fluid Dynamics Conference*, (Miami, FL), American Institute of Aeronautics and Astronautics, June 2007.

[74] Kufeld, R. M., Balough, D. L., Cross, J. L., Studebaker, K. F., Jennison, C. D., and Bousman, W. G., "Flight Testing of the UH-60A Airloads Aircraft," in *Proceedings of the 50th Annual Forum*, (Washington, D.C.), American Helicopter Society, May 1994.

[75] Bousman, W. G. and Kufeld, R. M., "UH-60A Airloads Catalog," Tech. Rep. TM 212827, NASA, 2005.

[76] Kufeld, R. M. and Bousman, W. G., "UH-60A Airloads Program Azimuth Reference Correction," *Journal of the American Helicopter Society*, vol. 50, no. 2, pp. 211–213, 2005.

[77] Bauchau, O. A. and Liu, H., "On the Modeling of Hydraulic Components in Rotorcraft Systems," *Journal of the American Helicopter Society*, vol. 51, no. 2, pp. 175–184, 2006.

[78] Brand, A. G., Komerath, N. M., and McMahon, H. M., "Wind Tunnel Data from a Rotor Wake/Airframe Interaction Study," Tech. Rep. DAAG29- 82-k-0094, Tech. Rep. CERWAT Report No., 1986.

[79] Brand, A. G., *An Experimental Investigation of the Interaction Between a Model Rotor and Airframe in Forward Flight*. PhD thesis, Georgia Institute of Technology, 1989.

[80] Liou, S. G., Komerath, N. M., and McMahon, H. M., "Velocity Measurements of Airframe Effects on a Rotor in Low Speed Forward Flight," *Journal of Aircraft*, vol. 26, no. 4, pp. 340–348, 1989.

[81] "http://www.ae.gatech.edu/labs/windtunl/expaero/jjht.html." Website, Last Accessed, May 2007.

[82] "http://windtunnels.larc.nasa.gov/facilities_updated/aerodynamics/14X22.htm." Website, Last Accessed, May 2007.

[83] Freeman, C. E. and Mineck, R. E., "Fuselage Surface Pressure Measurements of a Helicopter Wind-Tunnel Model with a 3.15-Meter Diameter Single Rotor," Tech. Rep. TM-80051, NASA, 1979.

[84] Ghee, T. A. and Elliott, J. W., "The Wake of a Small-Scale Rotor in Forward Flight Using Flow Visualization," *Journal of the American Helicopter Society*, vol. 40, no. 3, pp. 52–65, 1995.

[85] Mineck, R. E. and Althoff-Gorton, S. L., "Steady and Periodic Pressure Measurements on a Generic Helicopter Fuselage Model in the Presence of a Rotor," Tech. Rep. TM-210286, NASA, 2000.

[86] Smith, M. J., Potsdam, M., Wong, T., Baeder, J. D., and Phanse, S., "Evaluation of Computational Fluid Dynamics to Determine Two-Dimensional Airfoil Characteristics for Rotorcraft Applications," in *Proceedings of the 60th Annual Forum*, (Baltimore, MD), American Helicopter Society, June 2004.

[87] Park, Y. M., Nam, H. J., and Kwon, O. J., "Simulation of Unsteady Rotor-Fuselage Interactions Using Unstructured Adaptive Meshes," in *Proceedings of the 59th Annual Forum*, (Phoeinx, AZ), American Helicopter Society, May 2003.

[88] Hirsch, C., *Numerical Computation of Internal and External Flows Volume I: Fundamentals of Numerical Discretization*. New York: John Wiley & Sons Ltd., 1988.

[89] Bousman, W. and Norman, T., "Assessment of Predictive Capability of Aeromechanics Methods," in *AHS Specialist's Conference on Aeromechanics*, (San Francisco, CA), American Helicopter Society, January 2008.

VITA

Jennifer Nadine Abras was born in Baltimore, MD in 1980. She attended Roland Park Country School from pre-first through high school and graduated Cum Laude on June 1999. While there she was on the varsity badminton team, and began her research career working in the Laboratory of Fluid Dynamics on the submersible holography and holographic reconstruction project at Johns Hopkins University during the summer in 1998. She then attended Johns Hopkins University in Baltimore, MD pursuing a degree in Mechanical Engineering. She joined the fencing team for the first year before turning her focus to undergraduate research. She continued to work in the Laboratory of Experimental Fluid Dynamics over the summer semesters before starting to work during the fall and spring semesters after her first year. After three years she decided to pursue a research project which was closer to her interest in computational prediction and joined a new project in the Civil Engineering department working in the Computational Uncertainty Quantification area on stochastic finite element analysis. This new research job provided the opportunity to pursue a summer student internship at Sandia National Laboratories in Albuquerque, NM, where she continued to work in the area of uncertainty quantification. After four years she got her bachelor's degree in Mechanical Engineering, graduating with departmental and university honors in May 2003. While looking for a graduate program to fit her interests, which had always been focused on ultimately switching to Aerospace Engineering, she came across a description of the research being performed in the Computational Aeroelasticity Laboratory in the Aerospace Engineering Department of the Georgia Institute of Technology. She then joined this lab as a GRA, obtaining her master's degree in Aerospace Engineering in December 2005, while continuing on to finish her doctorate degree in May 2009. She currently works for NAVAIR in the applied aerodynamics branch in Patuxent River, MD.